

MICROHOBBY

AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

Semanal

AÑO II N.º 29

160 Ptas.

Canarias 165 pts.

**IRRESISTIBLE
ASCENSION
DE AMSTRAD**

**DISFRUTA
CON LAS
AVENTURAS
DE HAGGIE
EN LA GRUTA
ENCANTADA**

**Programar
en Basic
ya no es
necesario;
Bascomp
lo hace por ti**

SOFTWARE

**La espada y brujería
en disco ya está aquí.
Sorcery Plus te espera**



... Aún más sorpresas en el n.º 4

*ya está en
tu quiosco*

AMSTRAD

también disponible
para

SPECTRUM, PLUS, 128

COMMODORE 64

ENDZONE

Infiltrate en el sistema de seguridad del enemigo y destruye todos sus sistemas de comunicación. Cuidado con los radares de control, ya que al ser localizado disparan sobre ti.

AMSSNAKE

Tu amigo Snake está hambriento y se ha puesto en contacto contigo para que le ayudes a encontrar su comida. Procura esquivar las ratas que se crucen en tu camino.

DEATH

Conduce a tus amigos a través de caminos y junglas hasta encontrar las cuevas que te salvarán de los peligros. Cuidada con la isla dominada por la Araña Negra.

COLOR

Demuestra tu habilidad colocando las bolas del mismo color en las distintas columnas dispuestas para este fin.

DOBLEALT

Este programa es una utilidad que te permitirá obtener caracteres de doble altura en cualquiera de los modos de pantalla que posee el Amstrad.

795 pts.
(Incluido IVA)

YOUR

COMPUTER

Si no lo encontrara en su quiosco, solicítelo directamente a nuestra editorial.

ODG

Paseo de la Castellana, 268
28046 Madrid. Tel. (91) 733 25 99

La mejor selección de programas de juegos y utilidades, publicados en la revista de mayor difusión de ordenadores de Europa. Ahora reproducidos en cassette, en auténtica exclusiva mundial.

AMSTRAD

sumario

Año II • Número 29 • 18 al 24 de Marzo de 1986
160 ptas. (incluida I.V.A.)
Canarias, 155 ptas. + 10 ptas. sabretasa aérea
Ceuta y Melilla, 155 ptas.

Director Editorial

José I. Gómez-Centurion

Director Ejecutivo

José M.ª Díaz

Redactor Jefe

Juan José Martínez

Diseño gráfico

José Flores

Colaboradores

Javier Barcelo

David Sapuerta

Robert Chatwin

Eduardo Ruiz

Francisco Portala

Pedra Sudón

Miguel Sepúlveda

Francisco Martín

Jesus Alonso

Pedra S. Perez

Amalio Gómez

Secretaría Redacción

Carmen Santamaría

Fotografía

Carlos Condel

Javier Martínez

Portada

Manuel Barco

Ilustradores

J. Igual, J. Pons, F. L. Frontan

J. Septien, Peja, J. J. Mara

Edita

HOBBY PRESS, S.A.

Presidente

Maria Andrino

Consejero Delegado

José I. Gómez-Centurion

Marketing

Marta García

Jefe de Publicidad

Concha Gutiérrez

Publicidad Barcelona

José Galán Cortés

Tel: (93) 303 10 22/313 71 62

Secretaría de Dirección

Marisa Cogorro

Suscripciones

M.ª Rosa González

M.ª del Mar Calzada

Redacción, Administración y Publicidad

La Granja, 39

Poligono Industrial de Alcobendas

Tel.: 654 32 11

Telex: 49 480 HOPR

Dto. Circulación

Carlos Peropadre

Distribución

Coedis, S. A. Valencia, 245

Barcelona

Imprime

ROTEDEC, S. A. Crta. de Irún.

Km. 12,450 (MADRID)

Fotocomposición

Novocomp, S.A.

Nicolás Morales, 38-40

Fotomecánica

GROF

Ezequiel Salano, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos

de la revista

COMPUTING with the AMSTRAD

Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud America 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace necesariamente solidaria de las opiniones vertidas por sus colaboradores en los artículos firmados. Reservados todos los derechos.

Se solicitará control OJD

5 Primera plana

Las ampliaciones de memoria están aquí. Red local para el 6128 con disco duro.

Código máquina 15

Segunda parte de las instrucciones de control de la CPU. Interrupciones incluidas.

18 Las cuentas claras

En este artículo informamos al lector de la que se encuentra detrás del nombre de **Amstrad**, y las causas de su fulgurante éxito.

20 Serie Oro

Haggie es un juego de acción al que sólo se le puede calificar de increíble; parece mentira que no se trate de un programa comercial, y de que esté realizada prácticamente en Basic en su totalidad.

Nos cuenta la divertida historia del ayudante de Santa Claus. No decimos más para reservar a los lectores el placer de la sorpresa.

6 Primeros pasos

En el juego de comandos gráficos del **Amstrad Basic**, está incluido un grupo de ellos que nos permite realizar auténticos juegos malabares con el contenido de la pantalla. Aprendamos como se utilizan.



10 AmstradIdeas

Un interesante programa, creado por uno de nuestros lectores, demuestra la calidad de las gráficas en alta resolución de tu ordenador.

12 Mr. Joystick

Sorcey Plus es un juego de aventuras y acción, que, si estuviera plasmado en papel, se diría que es del género de **espada y brujería**.

Sin embargo, está en disco; esta le permite aprovecharse al máximo de la variedad y cantidad de pantallas, y de unos gráficos de gran belleza que muestran un estilo típico de los programas para **Amstrad**.



24 Análisis

Hace poco publicamos un análisis titulado **GENPANT**, un generador de pantallas. ¿Le gustaría que las pantallas creadas con el otro programa las convirtiera en Basic directamente ejecutable?

26 ProgramAcción

Alguién diga que la música es el más bello de los ruidos; después de leer este artículo, podremos afirmar que el ruido es la más bella de las músicas.

RAMBO

TM

FIRST BLOOD PART IITM



STALLONE

© 1985 Atlantic Systems NV. All rights reserved.
TM & Trade Dress of Atlantic Systems NV. Licensed by Warner J. Carter, Inc.

**OTRA
EXCLUSIVA**

ocean

ERBE

Software

SANTA ENGRACIA, 17. 28010 MADRID. Tel.: 447 34 10



RED LOCAL PARA EL 6128

Northen Computers ha creado una red local para el 6128. Esto quiere decir que una serie de 6128 puede intercambiar datos y programas, compartiendo los recursos de toda la red.

Concretamente, un 6128 actuando como «server» de la red hace funcionar un disco duro de 20 Megs y una impresora láser Canon; ambos periféricos pueden ser usados por el resto de la red, a la cual puede conectarse un ilimitado número de ordenadores a cualquier distancia.

Pero dicha red puede incorporar también ordenadores de otras marcas, como Apple, IBM PC, Apricot y BBC micros.

Sin embargo, por ahora al menos, el sistema cuesta algo caro: más de 150 £ (aprox. 35.000 ptas.) por ordenador y 1000 £ el disco duro (unas 230.000 ptas.).

De cualquier forma, esto abre el camino a multitud de aplicaciones, desde informatizar pequeñas empresas a bajo coste, hasta aulas informáticas y redes de 6128 en un edificio.

YA ESTAN AQUI LAS AMPLIACIONES DE MEMORIA

Como ya saben nuestros lectores, en la Amstrad Computer Show de Londres, muchas compañías presentaron importantes novedades en lo que a periféricos se refiere. Una de ellas, DK Tronics, mostró sus ampliaciones de memoria de 64 y 256 K para la serie CPC de Amstrad.

Estas memorias son de tipo lectura/escritura (RAM), y pueden conectarse juntas o por separado al ordenador, en el bus de expansión.

En definitiva, conectar ambos su-

pondría datar de una memoria RAM extra de 320 K a cualquier CPC.

Lo interesante es que ya se venden en España, a los siguientes precios:

— ampliación de 64 Kbytes: 15.350 (464/664).

— ampliación de 256 Kbytes: 37.000 (464/664); 34.375 (6128).

Las ampliaciones se pueden encontrar en:

Microware.

Clara del Rey, 58. Tel. 415 15 46.
28002 Madrid.

Primera PLANA

HARDCOPYS EN COLOR



Greysoft, otra empresa inglesa, ha desarrollado un programa para copiar en impresora pantallas, pero en color.

Greysoft clama que su obra es la primera de este estilo, y está pensando para funcionar con una impresora a Seikosha GP 700A.

Existe versión en disco y cassette.

Lamentablemente, no podemos daros los datos de Greysoft.

SOFTWARE DE MIRRORSOFT

Mirrorsoft ha lanzado en Inglaterra un programa de simulación de vuelo, llamado «SPITFIRE 40», en versiones para los ordenadores más conocidos del mercado de los «home computers», esto es, Amstrad (cinta/disco), Spectrum y Commodore.

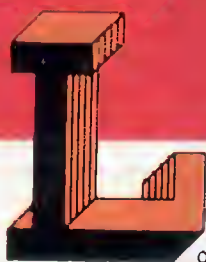
El juego, consiste en una mezcla de un simulador de vuelo con una aventura de acción y combate. Consta de varias partes: aprendizaje como piloto a través de varios grados de complejidad, dominio de las técnicas de vuelo y combate real.

Mirrorsoft Ltd., Maxwell House, 74 Workshop Street, London EC2A 2EN. Tel. 01-377 46 44.



DIBUJOS RELATIVOS

La pintura seguramente nació como una necesidad de experimentar con los colores, masas y formas para encontrar el mejor medio de expresar nuestros sentimientos internos por medio de ella. Estamos intentando ahora hacer algo parecido pero cambiando la paleta y los pinceles por un ordenador y un montón de buenas ideas. No es lo mismo, pero seguro que conseguirá magníficos efectos.



o primero que vamos a hacer es recordarle la forma como hemos preparado el «lienzo» donde realizaremos todos nuestros dibujos. ¡Animos!, seguramente que todavía la recuerda.

Para dibujar tenemos la pantalla dividida en 256.000 puntos (640x400) que podremos manejar individualmente con nuestros comandos. Es un buen número de puntos, ¿verdad? Fijese todas las cosas que podemos hacer con ellos.

El origen de estas coordenadas gráficas lo hemos colocado en el borde inferior izquierdo de la pantalla y por tanto irán creciendo hacia arriba —la «y»— y hacia la derecha la «x». Sus valores han de estar comprendidos entre 0 y 639 en la coordenada «horizontal» y de 0 a 399 para la «vertical» si queremos que el punto este situado en el centro de la pantalla.

Pero estos límites no son ningún obstáculo para que, si queremos, nos podamos colocar en cualquier sitio del espacio, «halaa».

Si, es cierto. La instrucción Basic que nos permite iluminar un punto es PLOT como vimos la semana pasada. Intente ver la que pasa al teclear:

PLOT 700,200

a

PLOT -45,200

Aparentemente no ha pasado nada. El ordenador ha admitido estas extrañas parámetros sin decir ni una palabra, pero en la pantalla no sale ningún punto. Y, ¿cómo va a salir si hemos elegido unas coordenadas que están situadas fuera de los límites?

Más adelante comprobaremos, que estos puntos «invisibles» por no estar en la pantalla tienen bastante utilidad, pero por el momento sólo le queremos indicar que tenga cuidado con ellas para no llevarse alguna «sorpresa» desagradable. El Amstrad no le va a decir nada si por error se dibuja un punto en «el más allá». Admite cualquier valor para las coordenadas gráficas, aunque sean negativas. Compruébelo con el Programa I.

Programa uno o cómo regresar del infinito

Con él vamos a traer un punto desde «casi» el infinito hasta el centro de la pantalla. ¿Cómo lo hacemos?

El proceso lógico que hemos seguido queda reflejado gráficamente en el organigrama, o diagrama de flujo, de la Figura 1.



Figura 1:
Organigrama
correspondiente al
programa I.



Veámoslo detenidamente. Primero situaremos nuestro «simple» punto en la posición inicial desde la que va a comenzar su viaje y damos aviso de que partimos. Va a ser la que tiene como coordenada 5000,200.

A continuación disminuimos la coordenada «x» (o sea, nos acercamos a nuestro objetivo).

El siguiente paso, es comprobar si hemos llegado ya al centro y actuar dependiendo del resultado que hayamos obtenido.

Si no estamos todavía en el punto de destino, o la coordenada «x» es distinta de 320, iluminamos el punto en las nuevas coordenadas y repetimos el proceso calculando un nuevo valor para la «x».

Si estamos ya en el centro, informamos de nuestra llegada y damos por terminado nuestro viaje.

Vamos a analizar la forma en que hemos plasmado este proceso lógico en instrucciones Basic, pero realizar con éxito este viaje imaginario.

La línea 30 fija el valor de la coordenada «x» del punto de partida en 5000. Un poco lejos, ¿verdad?

Lanzamos el mensaje de que nuestro viaje comienza con la línea 40 e iluminamos el punto en la 50. Como todavía está muy distanciado del centro de la pantalla, no advertiremos nada cuando el programa comience a ejecutarse.

A continuación restamos 1 a la coordenada «x», para acercarnos un poquito a nuestro destino mediante la asignación que hacemos en la línea 60. No olvide que esta instrucción no es una ecuación matemática en la que:

$$x = x - 1$$

nunca sería posible. Es, más bien, la forma de indicarle al ordenador, que primeramente efectúe el cálculo de lo que está a la derecha del signo igual y que, a continuación, coloque el resultado en el espacio que ha reservado en la memoria, para la variable que hemos llamado «x».

En la línea 70 es donde averiguamos si hemos llegado o no al centro de la pantalla.

La condición que ponemos es que la coordenada «x» sea distinta (\neq) o no de 320, que es el lugar donde hemos situado el centro de la pantalla.

Si se cumple, (es distinta) el programa saltará a la línea 50 y el proceso volverá a repetirse acercándonos un poquito más. Esto es exactamente lo que le decimos que haga con la instrucción:

GOTO 50

que hemos colocado entre el THEN y el ELSE de la sentencia IF.

De lo contrario (si no se cumple la condición— ELSE), es que ya hemos llegado y por tanto, después de anunciarlo con un mensaje, sólo nos falta terminar— END.

Imaginamos que le habrá resultado bastante fácil establecer un paralelismo entre cada una de las «cajitas» del diagrama de la Figura 1

Primeros PASOS

y las instrucciones Basic del programa que hemos desarrollado. Y también habrá podido comprobar que después de haber pensado y realizado el organigrama, seguramente le habrá sido mucho más fácil, ver qué es lo que tenía que hacer nuestro programa y cómo debía hacerlo.

El proceso lógico ya estaba decidido. Solamente faltaba reflejarlo en un lenguaje que entendiera el ordenador.

Después de esto, nos atrevemos a sugerirle que intente hacer todos sus programas partiendo de un organigrama, en el que estén ordenadas gráficamente, cada una de las acciones lógicas que debemos hacer para resolver nuestro problema.

Seguro que si ha confeccionado un diagrama de flujo detallado y correcto se encontrará con bastantes menos problemas a la hora de codificar sus programas. Háganos caso y ya verá cómo soldrá ganondo.

Todo es relativo. Nuestro Amstrad también

Volvamos con nuestros dibujos. Puede que en algunas ocasiones resulte engorroso andar calculando las posiciones (o coordenadas) de un punto de la pantalla. No olvide que estamos trabajando en una pantalla de 640 * 400 «pixels» —son muchos— y es muy fácil equivocarnos de sitio.

Pero, ¿qué le parece si vamos diciéndole a nuestro Amstrad que haga, o ejecute, una determinada instrucción ayudándose de las coordenadas relativas al último punto al que hemos llegado después de realizar alguna sentencia gráfica?

De momento nos encontramos con la ventaja de no tener que calcular la posición absoluta del nuevo punto, sino que nos bastaría con decir al ordenador: dibuja un punto que está 5 pixels a la derecha y 3 hacia arriba del anterior.

Y, ¿cómo decirle esto al Amstrad? No tengo miedo. Tenemos una buena máquina en casa, que va a ser capaz de hacerlo sin ningún problema, ya que está dotado de la herramienta que necesitamos.

Tecleando:

PLOT 5,3

tendríamos solucionada nuestra «pequeña» dificultad. No importa cual fuera el punto anterior. El micro dibujaría otro que está 5 más a su derecha y 3 más arriba, independientemente de su situación.

Pero no sólo es esta. Podemos asignar a estas coordenadas relativas valores negativos. Por ejemplo:

PLOTR —5,3

nos situará en un punto que estará a 5 pixels a la izquierda y 3 por encima del que hubiéramos dibujado la vez anterior. Y del mismo modo:

PLOTR 5,—3

iluminará uno que tenga por coordenadas las valares que resulten de sumarle 5 a la «x» y restarle 3 a la «y» del último que hayamos trazado. Estará, por tanto, más a la derecha y más abajo.

Observará que no existe ningún problema en darle cantidades negativas como parámetros a esta instrucción, lo que le da una gran utilidad y eficacia.

Veamos su forma más sencilla:

PLOTR x,y

nos dibujará un punto en la pantalla de gráficos en una posición cuyas coordenadas resultarán de incrementar en x (cantidad) la coordenada gráfica «x» actual y en un valor, y, la correspondiente coordenada «y». (Son las del último punto trazado mediante una instrucción PLOT o el final de una recta dibujada mediante DRAW.)

Esta es la forma más sencilla, pero podemos ir la complicando poco a poco.

Vuelva a colocarse en el centro de la pantalla mediante:

PLOT 320,200

y a continuación dibuje algunos puntos a su alrededor utilizando la instrucción PLOTR.

Poniendo a punto el color

Tecla ahora: PLOT 10,10,2

y nos aparecerá en la pantalla un maravilloso punto azul claro, colocado 10 posiciones más arriba y a la derecha del anterior.

Añadir un parámetro más a la instrucción PLOTR, ha tenido una bonita consecuencia: podemos cambiar a nuestro gusto el color de los pixels dentro de la gama que nos ofrezca el modo en el que estamos trabajando.

Recordará, o si no lo hacemos nosotros, que en Modo 1 (el que utilizamos habitualmente), solamente podemos tener disponibles cuatro colores diferentes, al mismo tiempo, en la pantalla.

El último parámetro que hemos añadido a la sentencia PLOTR, es el que determina el color de la tinta con la que va a salir el punto en la pantalla. En este caso y debido a que la tinta 2 está asignada al azul claro, el pixel se ilumina de ese color. Ya vamos dando vida, o por lo menos color a nuestras «obras maestras», ¿verdad?

Programa dos: ordenando el firmamento

Probemos hacer otro programa usando esta instrucción. ¿Se acuerdan del cielo estrellado, que se terminaba convirtiendo en un tremendo «sarampión», en nuestra artículo anterior?

Basándonos en él vamos a intentar que vayan apareciendo una serie de puntos alrededor del punto central de la pantalla, cada uno de un color, y que vayan manteniendo una cierta distancia respecto al anterior que hayamos dibujado.

El proceso lógico que vamos a seguir es el indicado por el organigrama de la Figura II. Veámosle.

Lo primero que hacemos es determinar las coordenadas del punto central de la pantalla e iluminarle con el color que está en curso. (Si acaba de inicializar el ordenador —RESET— estamos seguros de que le aparecerá de amarillo.)

Calculamos a continuación el color del siguiente punto que vamos a dibujar. Será uno, al azar, elegido entre las 4 que disponemos en este modo de trabajo.

Después calcularemos las coordenadas relativas del nuevo pixel, también al azar, partiendo de un algoritmo casero que nos fabricaremos para dar variedad al «asunto».

Y, por fin, trazamos el punto sin ningún problema. Observará que está muy cerquita del anterior tal y como habíamos especificado.

Vamos a repetir este proceso indefinidas veces, de ahí la flecha que une el último cuadro con el que nos calcula el nuevo color. Por tanto, el programa sólo se parará cuando pulsemos dos veces, por lo menos, la tecla Escape.

No habrá encontrada ninguna dificultad en seguirlo, «verdad».

Bueno, pues una vez que ya conocemos el proceso lógico que tenemos que realizar, vayamos a codificar el programa que haga toda lo especificada —Programa II.

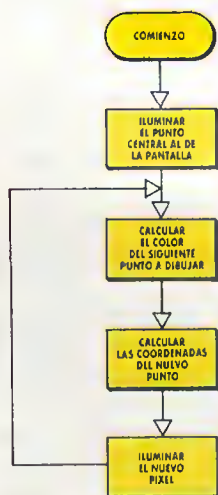


Figura II: Diagrama de flujo del programa II.

Colocamos un punto en el centro de la pantalla mediante la línea 30 que no ha de representarnos ningún problema. La coordenada «x» vale 320 y la y 200.

El ordenador elige un color al azar generando un número aleatorio comprendido entre 0 y 4 en la línea 40. Comprenderá el porqué de la utilización de la función INT en este caso.

De la misma manera determinaremos los valores de las coordenadas relativas del siguiente punto. En la línea 50 sacamos un número al azar que nos va a servir para fijarlas.

Si esa cantidad es igual a 0, el incremento de la «x» será positiva, mientras que la «y» va a mantener igual su valor —línea 60.

Podemos dejarle a usted que interprete cómo van a ser los incrementos que originan los distintos valores de la variable «azar». Es ya un programador «avanzado» y no debe detenerse ante estas cosas (líneas 70, 80 y 90).

Y por fin, iluminamos el nuevo punto, mediante la instrucción PLOTR de la línea 100. Dese cuenta de lo cerca que se encuentra del centro de la pantalla, sobre todo al principio. Solamente incrementamos o decrementamos sus coordenadas en 2 unidades.

Como no queremos que el proceso se detenga, vamos a enviarle a calcular el color y la nueva posición del siguiente punto mediante el oportuno GOTO de la línea 110.

Así estaríamos encerradas en un círculo del que sólo saldríamos con la ayuda de la tecla Escape, pero el cuadro que crearemos será «maravilloso».

En línea con la relatividad

Pero de la misma forma que hemos encontrado dentro de nuestro ordenador una herramienta que nos permite dibujar un punto sin conocer sus coordenadas absolutas, también posee una instrucción que traza una recta conociendo solamente la posición relativa del punto final de la misma respecto a la situación del último dibujado.

DRAWR 5,3

dibujará una recta cuyo contenido está en la última posición del cursor gráfico y que terminará 5 posiciones más a la derecha y 3 más arriba.

Con sólo dar al ordenador el desplazamiento relativo al último punto iluminado, él ya sabe hasta qué posición ha de trazar la recta.

Situémonos en el centro de la pantalla. Si tecla:

PLOT 320,200

estamos seguros que lo conseguiremos. Vamos a imaginar que queremos trazar una recta que nos devuelva al lugar de origen. Sería suficiente con hacer:

DRAWR —320,—200

y el Amstrad obedientemente nos lleva al sitio que queremos.

Mediante esta sentencia, es muy fácil dibujar una escalera que vaya subiendo por la pantalla y que comience en el origen de coordenadas. No encontrará ningún problema para hacerlo, ¿verdad?

El proceso lógico a seguir consistiría en situarnos en el origen, o mejor colocar allí nuestro cursor gráfico, y a partir de este punto, ir dando primero un incremento a la coordenada «y», trazar la recta que les une, y después hacer lo mismo con la «x», dibujando cada vez una línea entre los dos últimos puntos; que serán el origen y destino de la recta respectivamente.

Y, ¿hasta cuándo repetiremos el proceso? Pues si las coordenadas de alguna zona de nuestro dibujo llegan a los márgenes de la pantalla podemos decir nuestro trabajo por terminado. Comprendido, ¿no?

Por si acaso, le remitimos al diagrama de flujo de la Figura III y seguro que las cosas le quedarán bastante más claras.

¿Cómo lo codificaría, para que nos quedara un programa que funcione y que a la vez sea fácil de seguir?

Compare el organigrama anterior con el Programa III y verá que es más sencillo que lo que parece.

Programa tres: dibujando una escalera

Los puntos más importantes a tener en cuenta son:

Iluminar primero el origen de coordenadas, aunque de no hacerlo así el **Amstrad** lo asume y coloco el cursor de gráficos en este punto después de inicializar el ordenador. (Línea 30.)

A continuación trazamos una recta vertical —línea 40. Viendo que el incremento de la coordenada «x» es cero, podemos deducir inmediatamente que la recta trazada será paralela al eje «y».

Figura III: Organigrama del programa III.

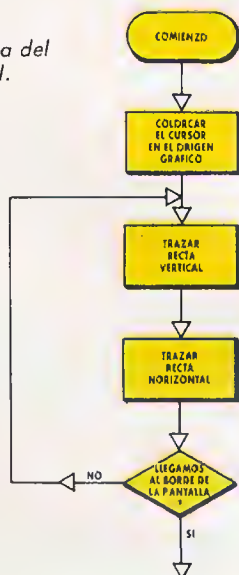
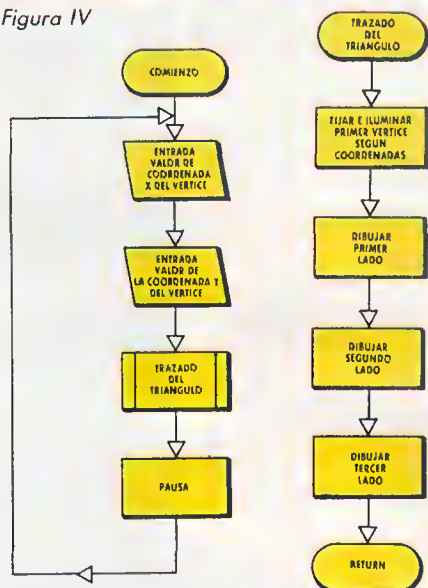


Figura IV



Organigrama programa principal.

Organigrama de la rutina que nos dibuja el triángulo.

Dibujamos después un segmento horizontal que nos une el punto donde se había quedado el cursor gráfico, con el punto calculado al incrementar la coordenada «x». La recta será paralela al eje «x» debido a que no incrementamos la coordenada «y» — en la línea 50.

Preguntamos si hemos llegado a los límites de la pantalla con el IF... THEN de la línea 60. Hemos utilizado el operador lógico OR para asegurarnos de que la línea no se nos va a escapar por ninguno de los bordes. Para comprobarlo utilizamos las funciones XPOS e YPOS, que nos dan las coordenadas horizontal y vertical respectivamente del cursor de los gráficos. Sin problemas ¿verdad?

Lo mismo que en el caso de PLOT, la sentencia DRAWR puede admitir también parámetros negativos para sus coordenadas.

Vayamos de nuevo al centro de la pantalla:

PLOT 320,200

Si tecleamos:

DRAWR -10,-20

aparecerá en la pantalla una recta que porte del punto de coordenadas 320,200 (centro) y que baja y va hacia la izquierda los valores de sus coordenadas van disminuyendo hasta llegar al punto 310,190. (Estos números resultan de sumar a la posición del centro los incrementos «negativos» que acompañan a la instrucción DRAWR).

Y respecto a dar color a la línea recta, podemos seguir las mismas directrices que dimos para la orden PLOT.

Si añadimos un parámetro más a DRAWR y tecleamos:

PLOT 320,200

seguido de

DRAWR 50,60,2

el **Amstrad** dibujará en la pantalla una recta de color azul claro que partirá desde el centro de la misma y se dirigirá a una posición que está 50 puntos a la derecha y 60 hacia arriba.

Programa cuatro: un problema triangular

Le proponemos un trabajo. Intente crear un pequeño programa que nos dibuje un triángulo en cualquier punto de la pantalla. Podemos aprovechar el Programa VI del artículo anterior. Con unas mínimas modificaciones lo conseguiremos.

Pero, una vez más, hemos decidido acordarnos de usted confeccionando el organigrama que representa gráficamente el proceso lógico que tendrá que seguir — Figura IV.

Y más todavía. Nos hemos tomado el atrevimiento de codificarle el programa en instrucciones Basic. Como siempre le recomendamos que primero intente hacerlo por sus propios medios y solamente utilice los que nosotros le proporcionamos en caso de que quiero comparar ambos o se encuentre en un atasco del que vea difícil la salida.

Observe que utilizamos una subrutina, hecha a base de las instrucciones que hoy les hemos especificado, para dibujar el triángulo. ¡Aplicación clara de toda la teoría que les hemos «soltado» sobre la programación realizada a base de módulos pequeños!. No le será difícil ver qué es lo que hace.

Y por si le queda alguna duda, eche un vistazo a la Figura V. En ella podemos resumir gráficamente todo lo que le hemos contado en este artículo. «x» e «y» son las coordenadas absolutas de un punto «xr» e «yr» las relativas de un segundo pixel respecto al primero.

¿Está mejor así?

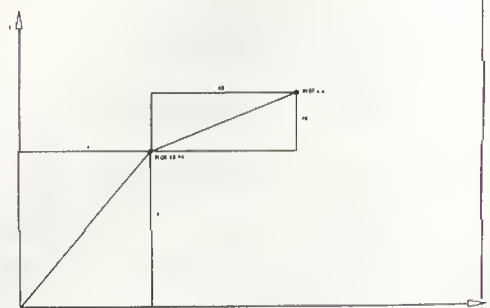


Figura V: Representación gráfica de las puntas iluminadas por PLOT x,y y PLOT XR,YR.

Vale por el momento. La próxima semana, nos dedicaremos a dibujar alguna casilla un poco más complicada, saliéndonos de las clásicas y ya más que conocidas líneas rectas. Esperamos contar de nuevo con su compañía.


```

10 REM PROGRAMA I
20 CLS
30 X=5000
40 PRINT"SALGO PARA EL CENTRO"
50 PLOT x,200
60 X=X-1
70 IF x<>320 THEN GOTO 50 ELSE PRINT"YA HE LLEGADO"

```

```

10 REM PROGRAMA II
20 CLS
30 PLOT 320,200
40 color=INT(RND*4)
50 azar=INT(RND*4)
60 IF azar=0 THEN x=2:y=0
70 IF azar=1 THEN x=-2:y=0
80 IF azar=2 THEN x=0:y=2
90 IF azar=3 THEN x=0:y=-2
100 PLOT x,y,color
110 GOTO 40

```

```

10 REM PROGRAMA III
20 CLS
30 PLOT 0,0
40 DRAW 0,10
50 DRAW 15,0
60 IF XPOS>640 OR YPOS>400 THEN PRINT"SE ACABO" ELSE GOTO 40

```

```

10 REM PROGRAMA IV
20 CLS
30 INPUT"COORDENADA X DEL VERTICE:"
  ,x
40 INPUT"COORDENADA Y DEL VERTICE:"
  ,y
50 GOSUB 1000
60 INPUT"PULSA ENTER PARA CONTINUAR"
  ,continuar$
70 GOTO 20
1000 REM TRAZADO DEL TRIANGULO
1010 PLOT x,y
1020 DRAW 200,0
1030 DRAW -100,INT(200*SQR(3)/2)
1040 DRAW -100,-INT(200*SQR(3)/2)
1050 RETURN

```

TORBELLINO

Este programa nos permite dibujar torbellinos y espirales de varios tipos. Cambiando un poco los bucles y los valores del seno puedes también realizar tus propios gráficos.

Jesús Manuel Lorenzo (Gijón)

El programa **Torbellino** es una pequeña utilidad que te permitirá representar gráficamente, con muy poco esfuerzo, una gran variedad de diseños gráficos basados en torbellinos y espirales. Podrás admirar, también, las magníficas capacidades gráficas de tu **Amstrad**, junto con la gran precisión que presentan los dibujos realizados en alta resolución.

Observa cómo alterando astutamente los valores de la función seno, y los índice de los bucles, podrás encontrarte en la pantalla con muchos dibujos distintos, y muy, pero que muy espectaculares.

Como ya he comentado antes, los diseños corresponden básicamente a torbellinos y espirales; los primeros, concretamente, se calculan y dibujan en las líneas 40-70 (la «salsa» en las líneas 50-60), y el resto del programa se dedica a tratar con las espirales, de las líneas 80-230.

A su vez, esta parte del programa se subdivide en tres tipos de dibujos: espiral normal, vista de canto y elíptica.

La primera se ejecuta en las líneas 120-150 (la importante en 140), la segunda va desde la 160 a la 190 (con el «meollo» en 180) y, la tercera, abarca las líneas 200-230 (clave: línea 220).

Una última cosa: merece la pena estar atento a la velocidad con que se desarrolla el programa, teniendo en cuenta los cálculos que tiene que realizar y que está escrito íntegramente en Basic.

```

10 REM TORBELLINO Y ESPIRAL A CHILE
1995
20 MODE 1:INPUT "MODO DE DIBUJO (1,2,3)
30 IF A=2 OR A=1 OR A=0 THEN MODE=A
40 PRINT"TORBELLINO":FOR A=1 TO 100
  STEP 1/4
50 ORIGIN 320,200+A
60 PLOT A/COS(A),A/3* SIN(A):NEXT
70 PRINT"PULSA UNA TECLA PARA CONTINUAR"
  :CALL AEB1B:CLS
80 PRINT"ESPIRALES":FOR A=1 TO 100
  STEP 1/4
90 ORIGIN 320,200
100 PLOT A/COS(A),A* SIN(A):NEXT
110 PRINT"PULSA UNA TECLA PARA CONTINUAR"
  :CALL AEB1B:CLS
120 PRINT"ESPIRAL NORMAL":FOR A=1 TO 75
  STEP 1/4
130 ORIGIN 320,200
140 PLOT A/COS(A),A* SIN(A):NEXT
150 PRINT"PULSA UNA TECLA PARA CONTINUAR"
  :CALL AEB1B:CLS
160 PRINT"ESPIRAL VISTA DE CANTO":FOR A=1 TO 75
  STEP 1/4
170 ORIGIN 320,200
180 PLOT A/COS(A),A/9* SIN(A):NEXT
190 PRINT"PULSA UNA TECLA PARA CONTINUAR"
  :CALL AEB1B:CLS
200 PRINT"ESPIRAL ELIPTICA":FOR A=1 TO 75
  STEP 1/4
210 ORIGIN 320,200
220 PLOT A/COS(A),A/3* SIN(A):NEXT
230 END

```


Si eres lector habitual de esta revista

¡llámame!
de 7 a 10 de la tarde



Te estoy esperando.

Tengo muchas cosas que contarte... y muy interesantes. De momento, te propongo la posibilidad de **AHORRAR** más de 1.000 ptas. y, además, con un poco de suerte, **GANAR UNA VESPINO** ¿Qué te parece? Pues esto es sólo un avance. Cuando me llames te contaré más cosas que seguro te gustarán.

Pero no te demores, porque a una mujer nunca se le hace esperar. Tienes de plazo hasta el 31 de marzo. Después, habrás perdido tu oportunidad.



HOBBY PRESS, S.A.

(91) 654 32 11

SORCERY PLUS

Aprender a dominar y utilizar las fuerzas ocultas de la naturaleza, es una de las tareas más difíciles en el duro aprendizaje del mundo de la hechicería.



En el reino de los brujos, donde todo era paz y sereno recogimiento, la vida de sus habitantes transcurría entre conjuros y hechicerías. Los ancianos enseñaban a los más jóvenes la forma de utilizar sus poderes y nada podía presagiar lo que se aproximaba.

El terrible Nigromante, dotado de poderes superiores, se apoderó de todos los brujos del país, encerrándolos en mazmorras, adueñándose de las riquezas y tesoros de todos.

Solamente un joven brujo fue capaz de escapar a las terribles artes de Nigromante, y con la sagrada misión de liberar a sus congéneres, emprende la dura lucha contra el gran mago.

Sorcery Plus, es un programa realizado exclusivamente para disco, razón por la cual en el paso de algunas pantallas a otras, veremos al ordenador leer datos del disc-drive.



Compatible: CPC/464, CPC/664 y CPC/6128



El uso del joystick es obligatorio, no pudiéndose realizar el juego por teclado.

Se trata de una típica aventura de búsqueda, en la cual debemos pasar por las distintas pantallas, liberando a los brujos presos, e intentando esquivar a las criaturas que, fieles al gran mago, evitan que cualquier intruso se acerque a los brujos presos.

Como en todas las aventuras de este tipo, encontraremos en nuestro paso por las distintas zonas de juego, diversos objetos cuyo uso adecuado es imprescindible para completar con éxito la misión.

La utilización de mapas para conocer la situación de los objetos clave y la situación que ocupamos dentro del marco geográfico del programa, puede ayudarnos en gran medida.

Todos los movimientos se consiguen con extrema facilidad, respondiendo a las distintas direcciones del joystick, con lo salvedad de que el brujo está sometido a la fuerza de la gravedad y descende por sí mismo, cosa a tener en cuenta en el salto sobre ríos y distintos obstáculos.

Las distintas tareas de recoger objetos, soltarlos, lanzar hechizos, etc., se realizan con una simple pulsación de disparo, simplificando la mecánica del juego.

Hay dos marcadores, que debemos vigilar con extremo cuidado; uno es el nivel de energía expresado en tanto por ciento, y el otro el tiempo.

La energía marca la cantidad de vida que nos resta para continuar el juego, y disminuye siempre que somos tocados por alguno de los elementos malignos que nos acosan en cada pantalla; cuando ésta alcanza el nivel cero, hemos fracasado en nuestra misión y debemos comenzar una nueva partida.

El tiempo del que disponemos para concluir la aventura, está marcado por un viejo libro de hechicería, el cual se va descomponiendo con el transcurso del mismo; cuando el libro desaparece reducido a polvo, habrá terminado el plazo marcado para nuestra meta.



Mister JOYSTICK



El tratamiento de los gráficos es vistoso y multicolor, estando éstos realizados en el modo de menor resolución con una acertada técnica de dibujos; los parajes con agua, murallas, zonas de bosque, etc., contribuyen a crear un juego variado y con creciente interés, la de intentar descubrir nuevos parajes. Las pantallas son planas, no utilizando sistemas de representación 3D, constando el entramado geográfico del programa de 75 localizaciones diferentes, en las cuales debemos realizar nuestra búsqueda.

El cambio de pantalla a pantalla se realiza instantáneamente, o con insignificantes lapsos de tiempo de retardo, cuando las nuevas pantallas han de leerse del disco, haciendo que casi no se note dicha lectura.

Un programa de larga duración, con diversidad de pantallas y un interés creciente, al descubrir la forma de abrir ciertas puertas que nos conducen a pantallas y zonas reservadas solamente a jugadores hábiles y tenaces.



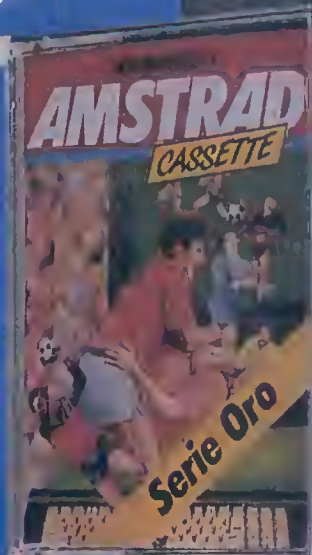
MICROHOBBY AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

Semanal

- Te ofrecemos un importante ahorro al realizar tu suscripción a Amstrad Cassette.
- Los mejores programas al mejor precio, totalmente desprotegidos con el objeto de facilitar su copia en disco.
- Cada cinta lleva grabados los programas publicados por Microhobby Amstrad durante cuatro números consecutivos (1 al 4, 5 al 8, 9 al 12, etc.)
- Compatibles con los modelos CPC-464, CPC-664 y CPC-6128.

¡AHORRA 1.500 PTAS.!



MICROHOBBY AMSTRAD

MICROHOBBY AMSTRAD

MICROHOBBY AMSTRAD

Recorta o copia este cupón y envíalo a Hobby Press, S. A. Apartado de Correos 232. Alcobendas (Madrid).

Deseo suscribirme a Amstrad Cassette durante un año (doce meses), al precio especial de 7.575 ptas., lo que me supone un ahorro de 1.500 ptas.

La primera cinta que deseo recibir es _____

Nombre _____ Edad _____

Apellidos _____

Domicilio _____

Ciudad _____ Provincia _____

C. Postal _____ Teléfono _____ Profesion _____

Marco con una (x) en el casillero correspondiente, la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre de Hobby Press, S. A.

☐ Giro postal n.º _____

☐ Contra reembolso del primer envío.

☐ Tarjeta de crédito VISA Número de la tarjeta _____

Fecha de caducidad de la tarjeta _____

☐ Domiciliación bancaria N.º de cuenta _____

Banco _____ Sucursal y Localidad _____

Fecha

Firma

INSTRUCCIONES DE USO GENERAL Y CONTROL DE LA CPU (II)

Realizar cálculos en código máquina puede resultar más sencillo de lo que siempre imaginamos. Junto con este tema, nos enfrentamos hoy con el control de interrupciones, tema que constituye sin posibilidad de duda la piedra angular del código máquina.



Empezamos la semana pasada la explicación de este bloque de instrucciones del cual hoy veremos las que nos restan, dando por terminada la explicación de este capítulo.

La instrucción que veremos en primer lugar se representa de la forma indicada:

DDA y SCF: suma y resta BCD

Esta instrucción ajusta condicionalmente el acumulador para operación de suma y resta en BCD (*Binario Codificado Decimal*).

Una vez ejecutada esta instrucción, el valor máximo que podrá contener el acumulador será de 99 decimal. Así pues, los cuatro bits inferiores podrán contener como máximo el valor 9 al igual que los cuatro bits superiores. Debemos hacer notar que el valor máximo que puede contener normalmente cualquier registro es de 256.

Para entender el efecto que produce dicha instrucción, en cada caso, hemos confeccionado la siguiente tabla, en la cual se indica la operación que se efectúa para la suma (ADD, ADC, INC) o la resta (SUB, SBC, DEC, NEG).

OPERACION	CARRY ANTES DAA	VALOR HEX EN BITS (7-4)	VALOR HEX EN BITS (3-0)	NUMERO SUMADO AL BYTE	CARRY DESPUES DAA
ADD ADC INC	0	0-9	0-9	00	0
	0	0-8	A-F	06	0
	0	0-9	0-3	06	0
	0	A-F	0-9	60	1
	0	9-F	A-F	66	1
	0	A-F	0-3	66	1
	0	0-2	0-9	60	1
	0	0-2	A-F	66	1
SUB SBC DEC NEG	1	0-3	0-3	66	0
	1	0-9	0-9	00	0
	0	0-8	6-F	FA	0
	0	7-F	0-9	A0	1
	1	6-F	6-F	9A	1
	1				

Vamos a ver un ejemplo para comprender mejor la actuación de la anterior instrucción. Supongamos que queremos realizar una operación de suma entre los valores 15 (BCD) y 26 (BCD) la aritmética simple nos daría como resultado el siguiente:

$$15 + 26 = 41$$

Pero cuando sumamos las representaciones binarias en el acumulador de acuerdo con la aritmética binaria, obtenemos lo siguiente:

$$\begin{array}{r} 00010101 \\ +00100110 \\ \hline 00111011 = 3B \end{array}$$

la suma obtenida no es la que deseábamos, ya que este valor lo obtenemos en hexadecimal. Así pues la instrucción DAA ajusta este resultado para obtener la representación correcta en binario codificado decimal:

$$\begin{array}{r} 00111011 \quad 3B \\ +00000110 \quad 06 \\ \hline 01000001 \quad 41 \end{array}$$

En el programa número 1 podemos ver un ejemplo práctico de actuación de dicha instrucción. Cargamos en

primer lugar el acumulador con un valor dado y a continuación ejecutamos DAA, y el resultado del acumulador lo colocamos en la dirección de memoria #7000 para poder inspeccionarlo desde Basic.

Otro ejemplo podemos observarlo en el programa número 2, en el cual efectuamos la misma operación, pero esta vez ponemos el flag Carry del registro F a 1 mediante la instrucción:

SCF

y a continuación ejecutamos DAA y preservamos el valor del acumulador en la dirección de memoria anteriormente mencionado para poder verlo desde Basic.

El programa número 3 nos muestra prácticamente el ejemplo expuesto anteriormente de la adición de dos números en BCD. En primer lugar debemos ajustar el acumulador con la instrucción DAA, a continuación cargamos en el acumulador el primer valor y luego le sumamos el valor que deseamos mediante la instrucción ADD. El valor obtenido es almacenado en la dirección de memoria #7000.

La instrucción con que nos encontramos a continuación se representa de la siguiente forma:

NOP: la estrategia de no hacer nada

Tras la ejecución de esta instrucción la CPU no realiza ninguna operación durante este ciclo de máquina.

Por lo tanto, el tiempo durante el cual no realizará ninguna operación será de 1 microsegundo, ya que éste es el tiempo que dura la ejecución de esta instrucción.

En el programa número 4 vemos un ejemplo práctico de aplicación de la instrucción NOP. Al llamar a este programa la CPU no realizará ninguna operación durante un microsegundo, y acto seguido volverá el control al programa Basic desde donde ha sido llamada la rutina.

El siguiente programa también utiliza la instrucción anterior, pero esta vez dentro de un bucle, por lo tanto el tiempo durante el cual no realizará ninguna instrucción será mucho mayor ya que dicha instrucción se repetirá durante 255 veces. Así pues el tiempo durante el cual no se realizarán operaciones será de 255 microsegundos.

Esta instrucción podría ser utilizada para crear pausas dentro de un bucle, en un programa en código máquina, para hacerlo más lento.

La instrucción que veremos a continuación se representa de la forma:

HALT

Esta instrucción suspende la operación de la CPU hasta que se reciba la siguiente interrupción o reset. Mientras está parado esperando una interrupción, el microprocesador ejecutará instrucciones NOP.

Así pues, ejecutando dicha instrucción, el procesador se parará hasta que se ejecute la siguiente interrupción.

Dado que el **Amstrad** ejecuta una serie de interrupciones, como por ejemplo leer el teclado, producción de sonidos, etc., entonces la instrucción HALT se deshabilitará cuando cualquiera de éstas sea llamada.

En el programa número 6 podemos ver un ejemplo práctico. En este caso ejecutamos la anterior instrucción, y cuando se encuentre con la siguiente interrupción, que puede ser alguna de las que hemos mencionado anteriormente, el programa retornará al Basic.

A continuación nos encontramos con la instrucción que se representa de la siguiente forma:

Hacia el control de las interrupciones

DI

Esta instrucción incapacita las interrupciones enmascarables durante su ejecución.

Cuando la CPU ejecuta DI, se incapacita la interrupción enmascarable hasta que vuelva a ser capacitada por una instrucción EI.

La CPU no responderá a una señal de petición de interrupción durante su ejecución.

Así pues, si en el transcurso de un programa en código máquina ejecutamos DI, a partir de ese momento

M. Barco



no se ejecutará ninguna de las interrupciones enmascarables, como la lectura de teclado etc.

En el programa número 7 podremos ver cómo actúa esta última instrucción.

En primer lugar deshabilitamos las interrupciones mediante DI, y a continuación ejecutamos la instrucción HALT. Como se puede suponer, esta rutina no retornará jamás al Basic, ya que al deshabilitar las interrupciones, no habrá ninguna que pare la ejecución de HALT por lo que el ordenador se quedará colgado en ese punto hasta que lo reseñemos.

La siguiente instrucción que veremos y que ya hemos visto anteriormente se representa de la siguiente forma:

EI

Esta instrucción permite que haya interrupciones enmascarables.

El activa las interrupciones enmascarables durante su ejecución.

Vamos a ver a continuación los distintos tipos de interrupciones que se pueden producir.

La primera de ellas se representa de la forma indicada a continuación:

Tres interrupciones para nuestro Z80

IM 0

Esta instrucción pone a uno el modo de interrupción 0. En este modo el dispositivo de interrupción puede insertar cualquier interrupción en el bus de datos y permite que la CPU lo ejecute.

La siguiente de estas instrucciones se representa de la forma:

IM 1

Esta instrucción pone a uno este modo de interrupciones. En este modo, el procesador responderá a una interrupción ejecutando un «restart» o vuelta a empezar.

La última de este tipo de instrucciones se representa de la forma siguiente:

IM 2

Esta instrucción pone a uno el modo de interrupción 2. Este modo permite una llamada indirecta a cualquier dirección de memoria.

En este modo la CPU forma una dirección de memoria de 16 bits. Los ocho bits superiores lo forman el contenido del registro I o vector de interrupción y los ocho bits inferiores se suministran por el dispositivo de interrupción.

Código MAQUINA

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00

10 ;PROGRAMA-1
20 ;
A000 30 ORG #A000
A000 3E03 40 LD A,99
A002 27 50 DAA
A003 320070 60 LD (#7000),A
A006 C9 70 RET

Pass 2 errors: 00
```

```
A000 00 40 NOP
A001 C9 50 RET

Pass 2 errors: 00
```

Table used: 13 from 105

Table used: 13 from 108

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00
```

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00

10 ;PROGRAMA-2
20 ;
A000 30 ORG #A000
A000 3E1D 40 LD A,29
A002 37 50 SCF
A003 27 60 DAA
A004 320070 70 LD (#7000),A
A007 C9 80 RET

Pass 2 errors: 00
```

```
10 ;PROGRAMA-5
20 ;
A000 30 ORG #A000
A000 06FF 40 LD B,255
A002 00 50 BUC: NOP
A003 10FD 60 DJNZ BUC
A005 C9 70 RET

Pass 2 errors: 00
```

```
BUC A002

Table used: 23 from 108
```

Table used: 13 from 109

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00
```

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00

10 ;PROGRAMA-3
20 ;
A000 30 ORG #A000
A000 27 40 DAA
A001 3E0F 50 LD A,15
A003 C61A 60 ADD A,26
A005 320070 70 LD (#7000),A
A008 C9 80 RET

Pass 2 errors: 00
```

```
10 ;PROGRAMA-6
20 ;
A000 30 ORG #A000
A000 76 40 HALT
A001 C9 50 RET

Pass 2 errors: 00
```

Table used: 13 from 105

Table used: 13 from 110

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00
```

```
Hisoft GENA3.1 Assembler. Page 1.
Pass 1 errors: 00

10 ;PROGRAMA-4
20 ;
A000 30 ORG #A000
```

```
10 ;PROGRAMA-7
20 ;
A000 30 ORG #A000
A000 F3 40 DI
A001 76 50 HALT
A002 C9 60 RET

Pass 2 errors: 00
```

Table used: 13 from 106

ALAN SUGAR Y LA IRRESISTIBLE ASCENSION DE AMSTRAD

Muchas dudas asaltan al usuario cuando llega el momento de elegir un ordenador, pero la fundamental es que detrás de su equipo, exista una infraestructura suficiente como para disponer en todo momento de asistencia y respaldo. La regla básica es muy simple: más ordenadores, más programas, más servicio. El presente artículo nos anima y tranquiliza. A partir de ahora las cuentas están claras.



Absolutamente brillante, un genio en su campo. Su acierto está en que ha sabido llegar al mercado en el momento justo, y salir de él con el mismo acierto».

Dijo Sir Clive Sinclair, un distinguido rival en el mercado de los microordenadores.

Desde el cuartel general de Computerland Essex, uno de sus directores ha declarado:

«Es agresivo, brillante y un hombre nacido para los negocios».

El hombre del que estamos hablando es Alan Sugar, fundador y director ejecutivo de **Amstrad**.

A la edad de 39 años, ha sorprendido a propios y extraños con los espectaculares resultados conseguidos en su gestión, al frente de la firma de ordenadores y productos electrónicos.

Poseedor del 54 por 100 de las acciones de la compañía, con su agresiva política comercial, ha visto elevar el valor de su parte de la empresa, de uno cifra de 27 millones y medio de libras, a un total de 188 millones.

Mientras las previsiones de los analistas, sobre los beneficios de **Amstrad** en los últimos seis meses de 1985, establecían un crecimiento del 60 por 100 alcanzando la cifra de 15 millones de libras, los resultados han triplicado estas expectativas, alcanzando una cifra de 27.5 millones.

Este incremento, que se sitúa 7.5 millones por encima de los resultados de todo el año

1984, hace que las previsiones para 1986 se sitúen en torno a los 50 millones de libras.

Incluso cuando **Alan Sugar** reconoce en sus declaraciones, que la demanda de ordenadores **Amstrad** particularmente en Francia, España y el Reino Unido, son más grandes que la cifra de producción disponible. Las cifras de venta de la compañía fuera del Reino Unido se sitúan en un 58 por 100 del volumen de negocio de la misma, esperándose que a finales del año en curso, ésta se sitúe en un 65 por 100.

Amstrad ha tenido el acierto de conseguir el éxito en mercados, donde ninguna peque-

ña firma británica, hubiera atrevido a aventurarse.

Hace solamente dos años, el 96 por 100 de su cifra de negocios estaba en la producción de aparatos de sonido, vídeo y televisiones, entrando en dura competencia con japoneses y demás países líderes en este sector. Seis meses después, **Amstrad** decidió abordar el mercado de los ordenadores domésticos, justamente cuando éste daba signos inequívocos de un alarmante descenso en la demanda.

Después a finales del pasado año, la compañía lanzó al mercado un ordenador personal para proceso de textos, cuando este tipo





de negocio estaba causando problemas a compañías mucho más grandes.

El éxito de **Amstrad**, viene de la forma en que la compañía concibe sus productos junto a sus precios excepcionalmente bajos. El mejor ejemplo, lo tenemos en el procesador de textos, que lanzada al mercado en el mes de septiembre, ha conseguido una cantidad de ventas, que le sitúan con el 21 por 100 de las ventas de **Amstrad** durante el último semestre de 1985.

El procesador de textos tiene un precio por debajo de las 130.000 pts, e incluye software, pantalla, disco e impresora. **Amstrad** ha sido la primera compañía en lanzar un ordenador de estas características, a bajo precio y concebido como una unidad, evitando de esta manera que el cliente tenga que comprar las diferentes piezas por separado y acoplarlas posteriormente. En palabras de Sugar **«Viene todo en una caja con un único cable, solamente hay que enchufarlo y empezar a trabajar».**

Su precio, es muy inferior a cualquier otro en el mercado y en palabras de un alto ejecutivo de Dixons: **«Nunca habíamos visto un producto como éste, los clientes lo ven y se muestran fuertemente inclinados a comprarlo».**

Los bajos precios de **Amstrad** se consiguen por los siguientes factores:

Al vender un sistema completo un gran número de componentes puede ser eliminado, en el caso del procesador de textos, éste tiene una sola fuente de alimentación y la impresora no tiene por qué ser compatible con una amplia gama de modelos. Las ampliaciones y demás aditamentos, que otras marcas utilizan para que sus modelos parezcan diferentes, no tienen objeto en **Amstrad**, Sugar argumenta que tales extras solamente son usados por muy pocos compradores.

Los ordenadores son fabricados en Corea, lo que significa un bajo costo de producción.

Amstrad posee una basta experiencia en la compra de componentes en el Oriente y una excelente red comercial. En lugar de llevar la misma política de compañías como Acorn y Sinclair Research, que atraviesan serias difi-

cultades, **Amstrad** prefiere contar con su bajo nivel de stocks de material.

La compañía tiene un número muy limitado de altos cargos, estando dirigida por un equipo muy reducido de personas, lo que hace que la mayoría de las decisiones, parezca que son tomadas en persona por el propio Sugar. Signo que es tomado por los analistas financieros como una debilidad, sobre todo ahora que la empresa se está haciendo mucho más grande.

La producción del procesador de textos, está alrededor de las 50.000 unidades al mes, cifra bastante prudente, a la que Sugar defiende con el argumento de que es mejor actuar con prudencia y no verse sorprendido después por un elevado número de máquinas en stock.

A pesar de que el procesador de textos, se espera que alcance un volumen de ventas del 35 por 100 a lo largo de todo el año, corren persistentes rumores de que **Amstrad** está preparando nuevos lanzamientos en los procesadores de textos y un compatible IBM. **Amstrad** reconoce que está trabajando en un IBM Pc compatible, que no será lanzado hasta el mes de julio, pero que sin embargo en un corto plazo de tiempo, va a tener lugar el nacimiento de una versión más potente del procesador de textos, con una memoria de 512 Kb y un disco duro probablemente.

A pesar del éxito conseguido en el campo de los ordenadores, la compañía pretende potenciar otros productos electrónicos, entre los que podemos destacar las cadenas musicales.

En este campo **Amstrad**, está a punto de fabricar dos nuevos sistemas de compact disc, los cuales llegarán al mercado en el mes de marzo.

Los precios de los dos modelos serán de 299 y 349 libras, estando integrados, por los siguientes elementos; pletina, amplificador, sintonizador de radio, cassette doble, equalizador gráfico y altavoces.

Después de la retirada de la compañía del sector de TV y el mercado del vídeo, Alan Sugar asegura: **«No debemos permanecer en un negocio donde los márgenes no son lo buenos que deseáramos».** A pesar de todo **Amstrad** ha empezado a vender vídeos de nuevo, entre ellos se anuncia el lanzamiento de un nuevo modelo con control remoto, al precio de 299 libras.

Sugar habla despectivamente de los riesgos que entraña el negocio de la alta fidelidad y los ordenadores personales, los cuales están basados en la tecnología.

El es en primer lugar un hombre de negocios, y la habilidad de la compañía que dirige, está en ensamblar productos en sistemas compactos, a precios bajos, la mayoría de los cuales son fabricados íntegramente en el lejano Oriente.

Los empleados de **Amstrad** en el Reino

Unido, alcanzan un total de 500, y solamente se dedican a ensamblar los equipos de sonido en la factoría de Essex.

Por otra parte, los ordenadores son fabricados en Corea, donde la empresa subcontratada por **Amstrad** tiene una plantilla de 2.000 empleados.

En palabras de Alan Sugar: **«Hace mucho tiempo ya quería que Amstrad fuera uno de los más grandes fabri-**

Informe Económico

cantes de aparatos electrónicos, estaba en un error, somos importadores. Ya hubiera estado contento de tener 2.000 empleados en Essex, pero los negocios son los negocios».

Entre los futuros planes de la compañía, también se encuentra el de vender productos electrónicos fuera del Reino Unido, aprovechando la popularidad que el nombre **Amstrad**, ha adquirido en el campo de los ordenadores, e intentando extenderla a toda la gama de productos que fabrica la firma.

Los costos de establecer nuevos acuerdos comerciales, para la distribución y venta de los nuevos productos serán elevados, y aún existe una total incertidumbre sobre la acogida que tendrá fuera del Reino Unido, el equipo de sonido básico que **Amstrad** quiere vender como avanzadilla.

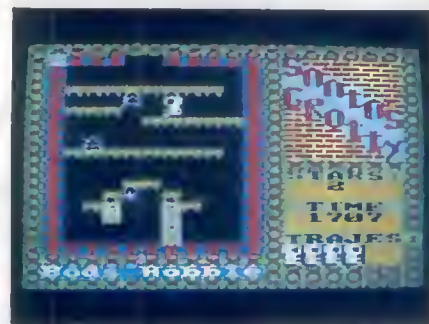
Con confianza en el éxito, y teniendo en cuenta que la fortuna le sonríe, **Amstrad** quizás acepte el riesgo que implica una empresa de estas características, aun más teniendo en cuenta que los fondos líquidos de la compañía a 31 de enero, se elevan a 23 millones de libras y que de esta forma es poco posible, que la firma tenga que recurrir a pedir dinero a sus distribuidores en un futuro próximo.



HAGGIE

Es difícil encontrar un programa que presente la brillantez y exquisitos gráficos del juego que hoy publicamos. Más de un programa comercial caería en un CRASH de vergüenza ante su simple presencia.

¡Anímate! y teclea, por que en esta ocasión merece verdaderamente la pena.



Estamos celebrando la noche de Navidad, y como cada año, tan pronto como llegó la oscuridad, Santa Claus salió para llevar bonitos regalos a todos los niños. En su prisa por empezar a trabajar, dejó caer unas cuantas estrellas por la plaza del pueblo.

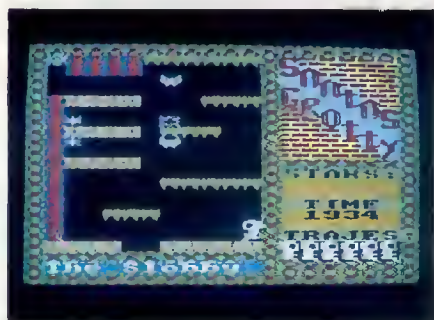
Esto poco normal, ha provocado que unos incordiantes monstruitos hallan salido de su oscuro escondrijo y roben las estrellas para venderlas en el mercado blanco. Tu tarea es bajar con **Haggy** —mano derecha de Santa Claus— a las cuevas de estos seres para recuperar las estrellas. Para hacer esto cuentas con tu trajejet, ten cuidado con las Meanies —estrellas moviéndose—, que pueden estropear. Tu misión debes terminarla antes de que caiga la bruma oscura y no puedas ser capaz de saber por dónde andas.

¡Buena suerte!. Vas a necesitarla...



EXPLICACION Y ACCION DEL PROGRAMA

10	REM con nombre del programa.
20	Modo 1 y selección de modos de pantalla.
30	Condición IF THEN para la determinación del modo elegido.
40	Nombre de la figura y bucle para dibujarlo.
50-60	Origen de coordenadas, dibujo con PLOT y fin de bucle.
70	Rutina del firmware para pasar pulsando uno tecla o dibujar espirales.
80-100	Nombre de la subrutina, bucle origen de coordenadas, dibujo con PLOT y fin de bucle.
110	Idem 70.
120-140	Nombre de subrutina, bucle, origen de coordenadas, dibujo con PLOT y fin de bucle.
150	Nombre subrutina y rutina del firmware para pulsar una tecla y pasar o dibujar la espiral hasta de canto.
160-180	Idem bucles anteriores, sólo cambia el final del bucle, el STEP o CONTADOR y el VALOR MULTIPLICADO por el SENOS DE A.
190	Idem 150.
200-220	Idem 160-180, cambia el STEP y el VALOR DEL SENOS para hacer ESPIRAL ELIPTICA.
230	END final del programa.




```

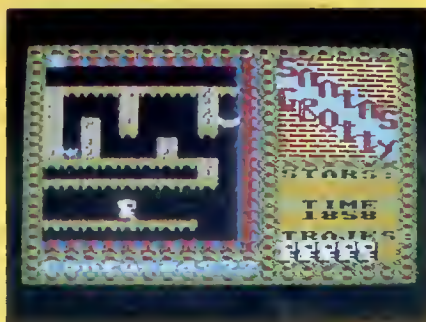
1 RE
M Santa's Grotty
2 REM By A.Chapman
3 REM Rutinas graficas
4 REM By R.A.Waddilove
5 REM (c) Amstrad Semanal
6 ON ERROR GOTO 30000
10 GOSUB 1790:REM U.D.G
20 GOSUB 2050:REM Inicializa
30 GOSUB 1650:REM Instrucciones
40 GOSUB 1480:REM Dibuja Border y P
antalla
50 GOSUB 1420:REM Variables
60 GOSUB 1180:REM Dibuja Pantalla
70 REM*****Rutina Principal*****
80 th=0
90 GOSUB 230
100 IF exit=1 THEN exit=0:GOTO 60
110 GOSUB 940
120 GOSUB 700
130 GOSUB 940
140 IF star=8 THEN POKE 34857,3:POKE
E 34868,3:IF screen=5 THEN LOCATE 1
1,2:PRINT " ":LOCATE 11,3:PRINT " "
150 IF star=19 THEN POKE 36297,3
160 IF star=28 THEN 990
170 IF lives<1 THEN 1040
180 ti=ti-1:IF ti<0 THEN ti=0
190 PAPER#1,14:PEN#1,0
200 IF ti>-1 THEN LOCATE #1,14,17:P
RINT#1,ti
210 IF st=1 THEN LOCATE #1,15,14:PA
PER#1,14:PEN#1,0:PRINT#1,star:st=0
220 GOTO 70
230 REM*****Mueva a Haggy*****
240 IF INKEY(71)<>0 AND INKEY(63)<>
0 AND INKEY(47)<>0 THEN GOTO 280
250 IF INKEY(71)=0 THEN GOSUB 400:G
OTO 280
260 IF INKEY(63)=0 THEN GOSUB 500:G
OTO 280
270 IF INKEY(47)=0 THEN GOSUB 600
280 IF j=19 THEN 330
290 IF y>16 THEN 310
300 GOSUB 690:IF PEEK(s+(po+22))=4
THEN star=star+1:POKE(s+(po+22)),3:
st=1
310 IF PEEK(s+(po+22))=3 THEN j=j+1
:po=po+11
320 IF j<20 THEN 360
330 IF loc(screen,2)=0 THEN 360
340 f=screen:screen=loc(screen,2):G
OSUB 690:IF PEEK(s+(x-1))<>3 OR PEE
K(s+(y-1))<>3 THEN screen=f:GOTO
360
350 po=i-1:i=2:exit=1
360 IF exit=1 THEN CALL &A000,x,y,x
,y,3:CALL &A000,x,y+1,x,y+1:GOTO
380
370 IF th=1 THEN CALL &A000,x,y,j,
,0:CALL &A000,x,y+1,j+1,pic+1:E
LSE IF th=0 THEN CALL &A000,x,y+1,j
,j+1,pic+1:CALL &A000,x,y+1,j,pic
380 x=i:y=j
390 RETURN
400 pic=6:IF i=2 THEN 450
410 SOUND 132,0,15,3,1,4
420 GOSUB 690:IF PEEK(s+po-1)=4 THE
N star=star+1:POKE(s+po-1),3:st=1
ELSE IF PEEK(s+po+10)=4 THEN star=s
tar+1:POKE(s+po+10),3:st=1
430 IF PEEK(s+po-1)<>3 OR PEEK(s+(p
o+10))<>3 THEN RETURN
440 i=i-1:po=po-1:RETURN
450 IF loc(screen,4)=0 THEN RETURN
460 f=screen:screen=loc(screen,4)
470 GOSUB 690:IF PEEK(s+(po+10))<>3
OR PEEK(s+(po+21))<>3 THEN screen=
f:RETURN
480 po=po-1
490 i=12:j=j-1:exit=1:RETURN
500 pic=4:IF i=12 THEN 550
510 SOUND 132,0,15,3,1,4
520 GOSUB 690:IF PEEK(s+po+1)=4 THE
N star=star+1:POKE(s+po+1),3:st=1
ELSE IF PEEK(s+(po+11))=4 THEN st
ar=star+1:POKE(s+(po+11)+1),3:st=1
530 IF PEEK(s+po+1)<>3 OR PEEK(s+(p
o+12))<>3 THEN RETURN
540 i=i+1:po=po+1:RETURN
550 IF loc(screen,3)=0 THEN RETURN
560 f=screen:screen=loc(screen,3)
570 GOSUB 690:IF PEEK(s+(po+10))<>3
OR PEEK(s+(po+11))<>3 THEN screen=f
:RETURN
580 po=po-10

```

```

590 i=2:exit=1:RETURN
600 SOUND 130,0,15,10,3,0,1:IF j=2
THEN 640
610 GOSUB 690:IF PEEK(s+(po-11))=4
THEN star=star+1:POKE(s+(po-11)),3:
st=1
620 IF PEEK(s+(po-11))<>3 THEN RETU
RN
630 j=j-2:po=po-22:th=1:RETURN
640 IF loc(screen,1)=0 THEN RETURN
650 f=screen:screen=loc(screen,1)
660 GOSUB 690:IF PEEK(s+(198+(x-1))
)<>3 OR PEEK(s+(187+(x-1))<>3 THEN
screen=f:RETURN
670 po=176+(i-1)
680 j=18:exit=1:RETURN
690 s=33999+((screen-1)*209):RETURN
700 REM*****Mueva Peligros*****
710 LET x=4*y1+1:4*y1=j:y1=h1-
h1:dir=dir1:11=111
720 ON dir1 GOSUB 860,880,900,920
730 CALL &A000,x1,y1,11,j1,gra1
740 x1=i1:y1=j1:dir1=dir
750 IF x2=0 THEN RETURN
760 LET x=2*y1+1:2*y1=j:y1=h1-
h1:dir=dir2:11=112
770 ON dir2 GOSUB 860,880,900,920
780 CALL &A000,x2,y2,11,j1,gra2
790 x2=i1:y2=j1:dir2=dir
800 IF x3=0 THEN RETURN
810 LET x=3*y1+1:3*y1=j:y1=h1-
h1:dir=dir3:11=113
820 ON dir3 GOSUB 860,880,900,920
830 CALL &A000,x3,y3,11,j1,gra3
840 x3=i1:y3=j1:dir3=dir
850 RETURN
860 i1=x+1:IF i1=h1 THEN dir=2:IF
screen=2 THEN j1=j+1 ELSE IF scree
n=11 OR screen=13 THEN i1=11:dir=1
870 RETURN
880 i1=x-1:IF i1=1 THEN dir=1:IF
screen=2 THEN j1=j-1
890 RETURN
900 j1=y+1:IF j1=h1 THEN dir=4
910 RETURN
920 j1=y-1:IF j1=1 THEN dir=3
930 RETURN
940 IF (x=x1 AND (y=y1 OR y+1=y1))
OR (x=x2 AND (y=y2 OR y+1=y2)) OR (
x=x3 AND (y=y3 OR y+1=y3)) THEN GOS
UB 960
950 RETURN
960 lives=lives-1:IF lives<1 THEN R
ETURN
970 OUT &BC00,8:OUT &BD00,1:SOUND 1
30,0,50,15,1,1,3:LOCATE #1,13+lives
,20:PAPER #1,14:PRINT#1," ":LOCATE
#1,13+lives,21:PRINT#1," ":OUT &BC0
0,8:OUT &BD00,0
980 RETURN
990 REM*****Felicitaciones*****
1000 RESTORE 1050:FOR f=1 TO 24:REA
D n:SOUND 4,n,20,15,1:NEXT
1010 LET score=(star*100)+(ti*2)
1020 PRINT STRING$(32,1):LOCATE 2,
7:PRINT"BIEN HECHO!":FOR f=1 TO 200
0:NEXT f:GOTO 2870
1030 DATA 60,53,47,45,60,0,45,47,45
,40,53,0,53,47,45,36,40,40,45,45,47
,53,47,60
1040 REM*****Fin del Juego*****
1050 LOCATE 1-1,j-1:PEN 3:PRINT CHR
$(276):LOCATE 1-1,j:PRINT CHR$(236)
1060 FOR f=1 TO 200:NEXT

```



Serie ORO

```

1070 SOUND 132,0,15,15,0,0,1
1080 LOCATE 1-1,j-1:PRINT CHR$(235)
:LOCATE 1-1,j:PRINT CHR$(235)
1090 SOUND 132,0,15,11,0,0,1
1100 FOR f=1 TO 200:NEXT
1110 LOCATE 1-1,j-1:PRINT CHR$(234)
:LOCATE 1-1,j:PRINT CHR$(234):SOUND
132,0,15,8,0,0,1
1120 FOR f=1 TO 200:NEXT
1130 LOCATE 1-1,j-1:PRINT " ":LOCATE
1-1,j:PRINT " ":FOR f=1 TO 200:NEXT
1140 FOR q=1 TO 2:RESTORE 1160:FOR
f=1 TO 11:READ d,n:SOUND 1,n,d,7:S
OUND 4,n+2,d,7:SOUND 5,0,3,0:NEXT f,
9
1150 LET score=(star*100)
+140:DATA 50,1016,37,1016,12,1016,5
0,1016,25,850,25,899,25,899,25,1016
,25,1016,25,1136,100,1016
1170 PRINT STRING$(40,1):LOCATE 2,
7:PRINT"GAME OVER":FOR f=1 TO 2000:
NEXT f:GOTO 2870
1180 REM*****Dibuja Pantalla*****
1190 PAPER 0:CLS
1200 IF ti=0 THEN GOTO 1240
1210 IF screen>6 AND screen<12 THEN
RESTORE 2340 ELSE RESTORE 2190
1220 FOR f=0 TO 159:READ n$:POKE &A
100+f,VAL("&"+n$):NEXT f
1230 LET s=33999+((screen-1)*209):F
OR f=2 TO 20:FOR g=2 TO 12:s=s+1:CA
LL &A000,g,f,g,f,PEEK(s):NEXT g,f
1240 LOCATE#1,2,22:PAPER#1,10:PEN#1
,4:PRINT#1,STRING$(11," "):LOCATE #
1,2,22:PRINT#1,sn$(screen):PEN 1
1250 RESTORE 2240
1260 FOR f=0 TO 31:READ n$:POKE &A1
80+f,VAL("&"+n$):NEXT f
1270 IF screen>10 THEN GOSUB 1310 E
LSE GOSUB 1300
1280 GOSUB 1340
1290 RETURN
1300 addr=&A100:RESTORE 2280:GOSUB
1320:addr=&A120:RESTORE 2290:GOSUB
1320:addr=&A140:RESTORE 2300:GOSUB
1320:RETURN
1310 addr=&A100:RESTORE 2310:GOSUB
1320:addr=&A120:RESTORE 2320:GOSUB
1320:addr=&A140:RESTORE 2330:GOSUB
1320:RETURN
1320 FOR f=0 TO 32:READ n$:POKE add
r+f,VAL("&"+n$):NEXT f
1330 RETURN
1340 no=(screen-1)*18-5
1350 y1=ene(no+(6)):x1=ene(no+(6)+1
)
1360 gra1=ene(no+(6)+2):dir1=ene(no
+(6)+3):l11=ene(no+(6)+4):h11=ene(n
o+(6)+5)
1370 y2=ene(no+(12)):x2=ene(no+(12)
+1)
1380 gra2=ene(no+(12)+2):dir2=ene(n
o+(12)+3):l12=ene(no+(12)+4):h12=en
e(no+(12)+5)
1390 y3=ene(no+(18)):x3=ene(no+(18)
+1)
1400 gra3=ene(no+(18)+2):dir3=ene(n
o+(18)+3):l13=ene(no+(18)+4):h13=en
e(no+(18)+5)
1410 RETURN
1420 REM*****Variables*****
1430 screen=1:lives=7
1440 x=6:y=15:i=6:j=15
1450 star=0:pic=4:score=0
1460 ti=2000:po=148
1470 RETURN
1480 REM*****Dibuja border*****
1490 MODE 0
1500 LOCATE 1,1:PAPER 1:PEN 3:PRINT
STRING$(20,CHR$(240))

```



```

1510 FOR f=2 TO 22:LOCATE 1,f:PRINT
CHR$(240):LOCATE 13,f:PRINT CHR$(2
40):LOCATE 20,f:PRINT CHR$(240):NEX
T
1520 LOCATE 1,23:PRINT STRING$(20,C
HR$(240)):LOCATE 1,21:PRINT STRING$
(12,CHR$(240)):LOCATE 14,22:PRINT S
TRING$(6,CHR$(240))
1530 PAPER 15:FOR f=2 TO 11:LOCATE
14,f:PRINT STRING$(6,CHR$(238)):NEX
T
1540 LOCATE 14,12:PAPER 1:PEN 3:PRI
NT STRING$(6,CHR$(240)):PAPER 2
1550 PEN 3:LOCATE 14,2:PRINT CHR$(2
41):LOCATE 14,3:PRINT CHR$(242):CHR$
(243):LOCATE 14,4:PRINT " "CHR$(244
)CHR$(245)
1560 LOCATE 14,5:PRINT CHR$(249) " "
CHR$(246)CHR$(247):LOCATE 14,6:PRIN
T CHR$(250)CHR$(251) " "CHR$(248)CHR
$(243):LOCATE 18,7:PRINT CHR$(244)C
HR$(241):LOCATE 19,8:PRINT CHR$(242
)
1570 LOCATE 15,7:PRINT CHR$(252)CHR
$(253) " " :LOCATE 16,8:PRINT CHR$(25
4)CHR$(247) " " :LOCATE 17,9:PRINT CH
R$(248)CHR$(247) " " :LOCATE 18,10:PR
INT CHR$(248)CHR$(255):LOCATE 19,11
:PRINT CHR$(239)
1580 FOR f=13 TO 21:PAPER 14:LOCATE
14,f:PRINT STRING$(6, " ") :NEXT:LOC
ATE 7,22:PAPER 10:PRINT STRING$(11,
" ")
1590 LOCATE 14,13:PAPER 1:PEN 3:PRI
NT"STARS:"
1600 LOCATE 14,19:PRINT"TRAJES:"
1610 LOCATE 14,16:PRINT" TIME "
1620 FOR f=1 TO 6:CALL &A000,13+f,2
0,13+f,20,4:CALL &A000,13+f,21,13+f
,21,5:NEXT
1630 WINDOW #1,2,12,2,20:WINDOW SWA
P 0,1
1640 RETURN
1650 REM*****Instrucciones****
***
1660 MODE 1:INK 14,15:INK 15,24:INK
1,18:INK 12,2,26
1670 LOCATE 1,1:PEN 1:PAPER 3:PRINT
STRING$(40,CHR$(240)):FOR f=2 TO 5
:LOCATE 1,f:PRINT CHR$(240):LOCATE
40,f:PRINT CHR$(240):NEXT f:LOCATE
1,6:PRINT STRING$(40,CHR$(240))
1680 LOCATE 15,7:PEN 3:PAPER 0:PRIN
T"SANTAS GROTTY":LOCATE 15,4:PEN 2:
PRINT STRING$(13,CHR$(131))
1690 LOCATE 1,8
1700 PEN 1:PRINT"Guia a Haggy a tra
ves de las cavernas de Santas Grott
y y recoge las estrellas que Santa
ha dejado tras el."
1710 PEN 3:PRINT"Cuidado con las qu
e se mueven, son muy peligrosas y p
ueden destruir tu traje cada vez q
ue choques con ellas."
1720 PEN 2:PRINT"Cuando el contador
de tiempo marque cero, una bruma o
scura descendera sobre la caverna d
ejandote sin vision."
1730 LOCATE 15,18:PEN 3:PRINT"LAS T
ECLAS":LOCATE 15,19:PEN 2:PRINT STR
ING$(10,CHR$(131))
1740 LOCATE 1,21:PEN 1:PRINT""Z"-IZ
QU. " :PEN 2:PRINT""SPACE""-SUB
IR " :PEN 3:PRINT""X"-DER."
1750 LOCATE 1,22:PAPER 3:PEN 1:PRIN
T STRING$(40,CHR$(240))
1760 LOCATE 1,23:PRINT CHR$(240):LO
CATE 40,23:PRINT CHR$(240):LOCATE 1
,24:PRINT STRING$(40,CHR$(240))
1770 LOCATE 5,23:PEN 3:PAPER 0:PRIN

```



```

T"Pulsa <ESPACIO> para continuar.."
1780 GOSUB 2650:GOTO 2870
1790 REM*****U.D.G*****
*
1795 ON ERROR GOTO 1805
1800 SYMBOL AFTER 230
1805 ON ERRDR GOTO 30000
1810 SYMBOL 240,66,165,90,60,60,90,
165,66
1820 SYMBOL 241,126,255,255,231,226
,224,224,254
1830 SYMBOL 242,255,127,7,71,231,25
5,255,126
1840 SYMBOL 243,24,60,176,255,231,2
31,231,231
1850 SYMBOL 244,255,231,231,231,231
,231,231,102
1860 SYMBOL 245,0,24,60,126,231,231
,231,231
1870 SYMBOL 246,231,231,231,231,231
,231,231,66

```

```

1880 SYMBOL 247,0,56,56,56,254,254,
56,56
1890 SYMBOL 248,56,56,56,56,57,63,6
2,60
1900 SYMBOL 249,126,255,255,230,224
,224,224,274
1910 SYMBOL 250,224,239,239,230,230
,254,254,124
1920 SYMBOL 251,62,127,231,231,271,
255,254,240
1930 SYMBOL 252,248,248,236,276,270
,230,227,227
1940 SYMBOL 253,24,60,126,271,271,2
31,231,231
1950 SYMBOL 254,231,231,231,231,231
,126,60,24
1960 SYMBOL 255,195,199,230,238,124
,60,28,28
1970 SYMBOL 239,28,28,56,48,112,96,
224,192
1980 SYMBOL 238,254,254,254,0,239,2
39,239,0
1990 SYMBOL 237,146,84,56,254,56,84
,146,0
2000 SYMBOL 236,144,70,16,132,66,20
,80,137
2010 SYMBOL 235,0,68,0,136,1,16,132
,1
2020 SYMBOL 234,8,0,34,0,0,8,65,0
2030 SYMBOL 233,198,165,198,165,6,4
0,40,16
2040 RETURN
2050 REM*****Inicializa*****
2060 INK 0,0:BORDER 0:PAPER 0:CLS:LO
CATE 10,10:PEN 2:PRINT"ESPERE POR
FAVOR!"
2070 RESTORE 2140
2080 MEMDRIY &84CF:check=0
2090 ENV 3,3,2,2,3,-2,2:ENT 4,5,-10
,20,1,0,5:ENT 1,30,10,1:ENV 1,10,-1
,2
2100 FOR f=0 TO 98
2110 READ n$:POKE &A000+f,VAL("&"+n
$)
2120 check=check+VAL("&"+n$)
2130 NEXT
2140 DATA dd,7e,00,87,87,87,87,87,3
2,47,a0,26,c0,dd,7e,04,3d,87,87,6f,
11,50,00,dd,46,02,05,19,10,fd,e5,26
,c0,dd
2150 DATA 7e,08,3d,87,87,6f,11,50,0
0,dd,46,06,05,19,10,fd,11,fd,07,06,
08,36,00,23,36,00,23,36,00,23,36,00
2160 DATA 19,10,f2,e1,11,00,a1,06,0
8,1a,77,13,23,1a,77,13,23,1a,77,13,
23,1a,77,13,78,01,fd,07,09,47,10,e9
,c9
2170 IF check<>7948 THEN LOCATE 10,
10:PRINT"ERROR EN DATAS":PRINT CHR$
(7):END
2180 RESTORE 2240:FOR f=0 TO 128:RE
AD n$:POKE &A100+f,VAL("&"+n$):NEXT
2190 DATA cd,cd,cd,cd,6a,6a,bf,95,b
f,95,bf,cd,bf,6a,6a,95,cd,95,6a,
cd,80,cd,80,40,00,40,80,40,00,00,80
2200 DATA cf,cf,cf,cf,ff,ca,cd,df,9
5,df,df,df,cd,cd,df,bf,9f,7f,ca,ca,
cf,cf,cf,cf,df,cd,9f,ea,df,7f,ca,df
2210 DATA 6e,cc,cc,9d,3d,cc,cc,3e,3
c,6e,9d,3c,3c,6e,9d,3c,3c,6e,9d,3c,
3c,6e,9d,3c,3d,cc,cc,3e,6e,cc,cc,9d
2220 DATA 00,00,00,00,00,00,00,00,0
0,00,00,00,00,00,00,00,00,00,00,00
2230 DATA 22,11,00,22,11,11,11,00,0
0,33,22,00,33,33,22,00,33,22,00,
11,11,11,00,22,11,00,22,00,00,00,00
2240 DATA 00,30,20,00,30,30,30,20,7
5,25,88,20,71,64,cc,20,75,64,88,20,
35,30,88,20,10,30,30,20,10,30,20,00
2250 DATA 10,71,20,00,3a,71,20,00,3
f,71,f3,e6,3f,30,20,00,3f,30,20,00,
3a,30,30,20,00,00,10,20,00,00,40,c0
2260 DATA 00,10,30,00,10,30,30,30,1
0,44,1a,ba,10,cc,98,b2,10,44,98,ba,
10,44,30,3a,10,30,30,20,00,10,30,20
2270 DATA 00,10,b2,20,00,10,b2,35,d
9,f3,b2,3f,00,10,30,3f,00,10,30,3f,
10,30,30,35,10,20,00,00,c0,80,00,00
2280 DATA 00,51,a2,00,00,f3,f3,00,5
1,ab,57,a2,3f,3a,35,1f,2f,0f,0f,2f,
05,8f,4f,0a,00,8a,45,00,00,8a,45,00
2290 DATA 08,04,08,04,08,04,08,04,0
4,aa,55,08,08,00,00,04,04,44,80,08,
04,08,04,08,0c,04,08,0c,00,00,00,04
2300 DATA 20,00,00,10,10,00,00,20,3
0,1a,25,30,25,0f,0f,1a,30,1a,25,30,
10,30,30,20,00,3f,3f,00,44,ee,dd,88

```

VARIABLES

X, Y	Coordenadas de Haggy
X1, Y1	
X2, Y2	Coordenadas de los Monstruos
X3, Y3	
gra1	
gra2	Tipos de monstruos
gra3	
li1, li1	
li2, li2	Límite mínimo y máximo del movimiento de los monstruos
li3, li3	
dir1, dir2	Dirección de los monstruos
dir3, dir4	1=Incrementa X, 2=Decrementa X 3=Incrementa Y, 4=Decrementa Y
star	Número de estrellas recogidas
score	Puntuación final según tiempo y estrellas recogidas
hs	Puntuaciones máximas
no\$	Nombres de las máximas puntuaciones
screen	Número de pantalla en juego
ti	Tiempo que resta para la bruma

SUBROUTINAS

70	Bucle principal
230	Mueve a Haggy
700	Mueve los peligros
990	Felicitaciones
1040	Fin del juego
1180	Dibuja pantalla
1420	Inicializa variables
1480	Dibuja border de pantalla
1650	Instrucciones
1790	Dibuja U.D.G
2050	Inicialización
2870	Max. Puntuación


```

2310 DATA 00,00,00,00,00,30,00,00,1
0,30,20,00,30,71,64,cc,30,30,64,cc,
10,30,20,00,00,30,00,00,00,30,20,00
2320 DATA 10,25,30,00,30,25,30,20,3
0,25,30,20,30,25,0f,20,30,30,20,
30,30,30,20,30,30,20,30,30,30,25
2330 DATA 40,c0,c0,80,80,00,00,40,c
c,88,44,cc,d5,10,00,ea,80,10,00,40,
80,10,30,40,95,00,00,40,95,3f,6a,c0
2340 DATA 0d,00,00,0e,0e,0a,05,0d,0
5,0d,0e,0a,00,0e,0d,00,00,0e,0d,00,
05,0d,0e,0a,0e,0a,05,0d,00,00,0e
2350 DATA 3f,3f,b7,3f,7b,7b,fb,3f,f
3,f3,f3,fb,51,5b,f3,a2,51,f3,fb,00,
00,fb,a2,00,00,51,00,00,00,00,00,00
2360 DATA 00,40,00,00,00,00,00,00,0
0,c0,80,00,00,c0,80,00,40,6a,6a,00,
40,90,c0,00,d5,d5,d5,40,cc,cc,cc,cc
2370 DATA 00,00,00,00,00,00,00,00,00,0
0,00,00,00,00,00,00,00,00,00,00,00,00
2380 DATA 22,11,00,22,11,11,11,00,0
0,33,22,00,3f,3f,3f,22,00,33,22,00,0
11,11,11,00,22,11,00,22,00,00,00,00
2390 RESTORE 2410
2400 no=33996:FOR f=1 TO 13:FOR g=0
TO 52:no=no+4:READ n:POKE no,n:64:
POKE no+1,(n AND 63):16:POKE no+2,(
n AND 15):4:POKE no+3,n AND 3:NEXT
g:no=no+3:NEXT f
2410 DATA 170,170,171,127,255,252,2
55,255,254,255,240,212,3,255,95,255
,252,67,255,255,42,63,255,63,255,25
5,255,255,255,240,42,175,255,255,25
2,3,255,255,255,255,255,255,85,255,
253,255,255,247,254,160,0,0,0
2420 DATA 42,128,3,170,255,255,255,
255,255,255,149,127,2,255,255,0
51,255,255,279,92,15,191,255,254,25
5,255,249,87,255,239,255,255,191,24
0,2,255,255,251,255,255,239,3,255,1
91,255,254,255,255,245,95,85,86
2430 DATA 170,250,170,255,255,235,2
55,255,160,0,14,191,255,250,3,255,2
35,252,0,175,255,254,191,255,250,0,
0,235,255,255,175,255,254,191,15,25
0,243,207,235,31,79,175,125,254,191
,247,250,255,223,234,170,170,168
2440 DATA 128,0,2,255,255,248,255,2
43,239,51,255,188,245,254,213,85,91
,255,255,239,255,255,181,85,94,223,
255,123,127,253,236,255,243,188,255
,62,252,243,251,255,255,237,255,247
,191,125,254,255,255,250,170,170,18
6
2450 DATA 170,170,171,255,255,239,2
55,255,144,16,18,127,127,121,253,25
3,231,119,247,157,207,206,119,247,2
49,223,223,224,0,7,191,255,222,0,0,
59,255,255,239,255,255,191,255,254,
0,0,251,255,255,234,170,170,186
2460 DATA 170,170,171,253,85,95,240
,0,47,255,255,179,255,254,252,0,27,
255,255,99,255,253,191,64,2,253,255
,251,55,255,239,16,16,191,63,62,63,
255,251,255,127,111,192,0,191,255,2
46,255,255,216,0,48,53
2470 DATA 85,87,87,255,255,207,255,
255,63,255,252,254,171,243,240,15,2
07,207,63,63,60,252,255,255,243,207
,255,207,254,255,48,0,252,255,255,2
43,243,255,207,255,255,63,63,252,25
5,255,242,170,170,128,0,0,48
2480 DATA 0,0,3,255,255,255,255,255
,234,175,254,255,255,247,255,255,24
5,95,255,220,255,254,252,255,247,25
2,255,252,252,255,240,252,250,112,2
52,195,255,255,15,255,188,63,236,0,
255,48,0,236,192,0,0,0,48
2490 DATA 0,0,0,255,255,243,255,255
,207,252,0,63,243,252,255,207,243,2
53,85,79,255,255,63,255,252,255,170
,163,255,255,207,255,255,63,213,84,
255,255,243,235,255,207,3,59,48,0,1
2,0,0,0,0,0,63

```

LAS TECLAS

Z Izquierda
X Derecha
Espacio ... Sube

```

2500 DATA 255,255,51,255,252,206,17
4,171,48,48,12,192,240,49,3,192,192
,7,3,0,60,12,85,245,115,255,252,207
,255,3,0,0,257,255,255,243,255,254,
207,162,135,63,95,252,255,255,243,1
91,238,192,15,0,48
2510 DATA 255,63,255,252,255,252,3,
0,51,255,255,207,255,255,255,255,25
5,245,87,255,255,255,87,255,255,255,
,255,255,255,255,255,255,191,255,25
1,191,255,188,255,251,254,245,79,24
3,255,255,247,255,255,197,85,85,107
2520 DATA 255,255,257,95,245,95,192
,15,255,247,255,252,0,255,251,103,2
55,255,255,255,255,255,252,0,255,25
5,255,255,255,255,213,127,255,255,2
47,127,255,93,255,255,117,255,196,2
15,247,207,79,95,253,212,48,195,67,
2530 DATA 255,255,255,255,255,255,2
55,255,255,240,0,127,255,254,127,25
5,255,124,63,253,255,255,189,255,25
4,247,195,251,7,255,239,223,240,191
,223,254,223,127,250,177,255,279,24
3,255,191,255,254,223,255,208,0,0,0
2540 DIM ene(234):RESTORE 2550:FOR
f=1 TO 234:READ ene(f):NEXT f
2550 DATA 11,4,0,1,2,11,3,6,0,1,4,1
2,14,4,1,3,14,19,4,8,2,1,4,12,8,8,1
,2,6,12,10,8,2,1,4,12,6,5,1,1,3,10,
10,5,1,2,3,10,0,0,0,0,0,0,10,3,0,3,
5,19,10,12,0,4,8,19,16,5,1,1,4,9,10
,11,2,4,6,19,0,0,0,0,0,0,0,0,0,0,0,
0,5,5,1,3,3,12,19,3,0,1,3,10,0,0,0,
0,0,0
2560 DATA 7,3,0,3,3,12,10,10,0,1,2,
11,0,0,0,0,0,0,4,5,2,1,2,12,6,9,2,2
,2,10,10,10,2,3,3,15,3,5,0,1,3,10,9
,5,0,2,3,12,12,5,0,1,3,10,2,5,2,2,2
,9,11,4,1,1,2,9,10,11,2,3,2,19,18,5
,0,1,5,11,19,5,1,1,5,11,7,7,2,1,2,1
2,9,7,2,1,3,11,11,7,2,2,2,12,0,0,0,
0,0,0
2570 DATA 18,5,0,1,5,12,19,5,1,1,5,
12,4,2,2,1,2,12
2580 RESTORE 2590:DIM sn$(15),loc(1
5),na$(8),hs(8),x1(3),x2(3),dir(3
),gra(3),11(3),h1(3),j1(3),11(3):FO
R f=1 TO 15:READ sn$(f):NEXT
2590 DATA Haggies Pit,The Slobby,Bo
ds Hobbie,The Sewer,Confused???,Hel
lo World,Low down,The Fitts,The Thri
ps,BiambaBon,The Eggloo,Mt.Chapman,
Santa's Hut
2600 RESTORE 2610:FOR f=1 TO 15:FOR
g=1 TO 4:READ loc(f,g):NEXT g,f
2610 DATA 0,0,5,2,0,3,1,4,2,0,0,0,0
,0,2,0,0,0,6,1,0,7,10,5,6,0,0,8,0,0
,7,9,0,0,8,0,11,3,0,6,12,10,0,13,0,
11,0,0,0,0,11,0
2620 FOR f=1 TO 8:na$(f)="BIG SCORE
S":hs(f)=3500-(400*f):NEXT
2630 te=0.6:0=1
2640 RETURN
2650 RESTORE 2800
2660 READ p,d:IF p=999 THEN RESTORE
2800:GOTO 2660
2670 GOSUB 2770
2680 SOUND 1,pn,d#0.5,15,1
2690 p=100:GOSUB 2770
2700 SOUND 1,pn,5,15,1
2710 WHILE INKEY$<>"":WEND
2720 IF INKEY(47)<>0 THEN GOTO 2660
2730 RESTORE 2960:FOR f=1 TO 28:REA
D g,f:1:POKE 73999+(g-1)*209:(f1,4:
NEXT f
2740 POKE 74857,2:POKE 74868,2:POKE
74297,2
2750 LOCATE 1,1:PRINT STRING$(26,11
)
2760 RETURN
2770 fr=440*(2*(0+((p-10)/12)))
2780 pn=ROUND(125000/fr)
2790 RETURN
2800 DATA 5,40,5,40,12,40,12,40,10,
40,8,40,7,40,5,40,3,40,5,40
2810 DATA 7,40,8,40,10,40,12,80,5,4
0,5,40,12,40,12,40,10,40,8,40,7,40,
5,40,3,40,5,40,7,40,8,40,10,40
2820 DATA 12,80,12,40,13,40,10,40,1
2,40,13,40,15,40,17,40,12,40,10,40
2830 DATA 8,40,5,40,7,40,8,40,10,80
,8,40,10,40,12,80,13,40,12,40,12,40
,10,40,8,40,7,40,5,80,8,20,7,20,5,4
0
2840 DATA 10,80,8,40,10,40,12,40,13
,40,15,40,17,40,12,40,10,40,8,40,7,
40,5,80
2850 DATA 999,999

```



```

2860 DATA 1,59,1,110,2,68,2,90,3,14
8,4,104,4,107,5,101,6,76,6,105,7,71
,7,72,8,157,8,158,8,151,9,52,9,165,
9,154,10,166,11,178,11,140,12,33,12
,23,12,60,12,62,13,9,13,67,13,68
2870 REM*****High Score*****
*
2880 MODE 1
2890 LOCATE 4,5:FEN 3:PRINT CHR$(15
0):STRING$(32,CHR$(154)):CHR$(156)
2900 LOCATE 16,3:FEN 2:PRINT"HIGH S
CORE":FEN 3
2910 FOR f=6A TO 15:LOCATE 4,f:PRINT
CHR$(149):LOCATE 37,f:PRINT CHR$(1
49):NEXT
2920 LOCATE 4,16:PRINT CHR$(147):ST
RING$(32,CHR$(154)):CHR$(153)
2930 FOR f=1 TO 8
2940 IF score>hs(f) THEN GOSUB 3030
:f=10
2950 NEXT
2960 FOR f=1 TO 8:FEN 1:LOCATE 8,f+
6:PRINT na$(f):LOCATE 18,f+6:FEN 3:
PRINT".....":hs(f):NEXT f
2970 LOCATE 1,17:PRINT STRING$(220,
" ")
2980 IF INKEY$>" " THEN GOTO 2980
2990 FEN 2:LOCATE 1,20:PRINT STRING
$(40,CHR$(154)):LOCATE 1,27:PRINT S
TRING$(40,CHR$(154)):FEN 1
3000 LOCATE 1,23:PRINT STRING$(40,"
")
3010 LOCATE 4,21:FEN 3:PRINT"FULSA"
:FEN 1:PRINT" ESPACIO ":FEN 3:PR
INT"PARA JUGAR.":FEN 1
3020 GOSUB 2650:DOTG 40
3030 IF "ABCDEFGHIJLMNOPQRSTUVWXYZ
0123456789+-*/<>~!"CHR$(255)
3040 c=19:LOCATE 1,20:FEN 1:PRINT a
$
3050 LOCATE 3,17:FEN 1:PRINT" TECLA
S 120U.DER. Y COPY DEL CURSOR ":LO
CATE 4,18:PRINT"PARA SELECCIONAR LF
TRAS. (MAX. DE 10.):":FEN 2:LOCATE 1,
19:PRINT STRING$(40,CHR$(154)):LOCA
TE 1,22:PRINT STRING$(40,CHR$(154))
3060 LOCATE 9,23:FEN 3:PRINT"PULSE"
:FEN 1:PRINT"< X >":FEN 3:PRINT
"PARA SALIR.":FEN 1
3070 x$=""
3080 FOR z=1 TO 10
3090 LOCATE c,21:FEN 2:PRINT" "
3100 IF INKEY(1)=0 AND c<40 THEN c=
c+1
3110 IF INKEY(8)=0 AND c>1 THEN c=c
-1
3120 IF INKEY(9)=0 AND c=40 THEN LO
CATE 7,f+6:PRINT" ":z=100
:GOTO 3170
3130 IF INKEY(63)=0 THEN z=11:GOTO
3170
3140 IF INKEY(9)<>0 THEN LOCATE c,2
1:PRINT"*":FOR a=1 TO 50:NEXT:GOTO
3090
3150 x$=x$+MID$(a$,c,1):LOCATE 7+c,
f+6:FEN 1:PRINT MID$(a$,c,1)
3160 FOR a=1 TO 200:NEXT
3170 NEXT
3180 IF z=101 THEN GOTO 3070
3190 hs(8)=score:na$(8)=x$
3200 f=0
3210 FOR z=1 TO 7
3220 IF hs(z)<hs(11) THEN t=hs(z+1
):hs(z+1)=hs(z):hs(z)=t:a$=na$(z+1)
:na$(z+1)=na$(z):na$(z)=a$:f=f+1
3230 NEXT
3240 IF f=1 THEN GOTO 3200
3250 fr=FRE("")
3260 RETURN
30000 MODE 1
30001 INK 1,24:PAFER 0:FEN 1:PRINT
"Error...":ERR:"at line":ERL
30002 END

```


Análisis

BASCOMP

¿Se acuerdan de GENPAN, el programa de análisis del número anterior? La idea en esta ocasión es la misma, editar una pantalla de texto y poder utilizarla en sus propios programas. BASCOMP lo que hará con el texto será traducirlo a Basic. Increíble pero cierto.

P

rimero se le pedirá el modo de pantalla 1, para 40 columnas, y 2, para 80. Ahora espere unos segundos y edite su pantalla igual a como lo hacía con GENPAN; cuando haya finalizado su edición pulse la tecla ENTER. Se le pedirá el nombre con el que quiere archivar el programa que generará la pantalla. Una vez tecleado este nombre, la pantalla comenzará a compilarse, a la vez que se va grabando en la cinta o disco. Finalizada la compilación haga un reset y cargue mediante LOAD el programa que ha compilado con el nombre elegido. Y ¡voilà!.

No entraremos en detalles sobre el programa en general, pues es igual a GENPAN. Lo que si veremos en detalle es como pone los caracteres de la pantalla en forma de programa.

10-30: Establecemos el modo de pantalla.

40-160: Subrutinas del editor.

170-270: Editor.

290: Borrarnos la pantalla y se nos solicita el nombre con el que vamos a grabar el programa, que vamos a crear.

300: Abrimos el fichero nom\$, es decir, el nombre que nosotros habíamos elegido en la línea anterior.

310: La variable n1 va a ser la encargada de controlar los números de línea con que vamos a ir almacenando las líneas de nuestro programa. Le damos el valor 1000, esta será la primera línea del programa.

320: Escribimos en el fichero el número de línea, la palabra mode y el número que contiene la variable modo. Obsérvese que aquellos valores que no son numéricos, por ejemplo, n1, los pasamos a cadena de caracteres mediante la función STR\$. El hecho de utilizar PRINT en vez de WRITE, se basa en que el

WRITE utiliza comillas como separador de datos y esto nos impediría crear un programa basic.

330-370: Bucle en el que se miran todos los elementos de la tabla, leyendo de izquierda a derecha y por filas de arriba a abajo. Si encontramos en una fila, un carácter distinto del espacio blanco, enviamos la ejecución del programa a la línea 400. La razón está en no editar líneas compuestas sólo por blancos.

Subrutina compila línea:

410: Incrementamos el número de línea, n1, en 10.

420: Creamos la variable lin\$, cuyo contenido será: el número de línea actual, locate, la columna y la fila correspondiente; fi y co; ;,print y finalmente chr\$(34), que corresponde al símbolo de los comillas.

430: Ahora desde esta columna vamos hasta el final de la línea, leyendo todos los caracteres, que se van añadiendo a la variable lin\$, en la línea 440.

470-490: Lo que aquí hacemos es suprimir de la variable lin\$, todos los caracteres blancos, que existan en el lado derecho, la razón es puramente económica. El sistema es muy sencillo, vamos preguntando por el último carácter de lin\$, línea 480, y si es un carácter blanco lo suprimimos y repetimos el proceso, de no ser así, damos el proceso por finalizado.

500: Escribimos en el disco o cinta, la variable lin\$. Como ya habrá observado, esta variable constituye por sí sola una verdadera línea del programa que nuestro CPC está realizando.

510-520: Hacemos regresar la ejecución al bucle de la línea 410, no sin antes cargar co, la columna, el valor más alto que puede alcanzar; para de este modo advertir a nuestro CPC que debe comenzar a compilar una nueva fila.

Quizá todo esto le haya costado un poco, no se desanime, el problema no está en la complejidad del programa sino en el efecto osombroso que produce. Estúdielo con paciencia y no lo olvide si hay dudo... a SIN DUDA ALGUNA.



P ara que tus dedos no realicen el trabajo duro, M.H. AMSTRAD lo hace por ti. Todos los listados que incluyen este logotipo se encuentran a tu disposición en un cassette mensual, solicítalos.



Te ofrecemos algo "muy Especial"

En el mes de septiembre nació una **AMSTRELLA** que vino a demostrar que los **Amstrad** estaban ahí y había que contar con ellos.

Hoy, miles de personas nos dan la razón.

Por este motivo, y después de los 6 primeros meses de andadura juntos con nuestros lectores, ha vuelto a suceder algo muy importante: ha nacido una **AMSTRELLA MUY ESPECIAL**.

¡Ya está a la venta en tu quiosco!
por sólo **350** ptas.



HOBBY PRESS
Para gente inquieta.

RUIDOS

Seguramente siempre soñó con aderezar ese jueguito casero —al que dedicó tantas tardes y horas de sueño— con disparos láser, despegues con los reactores a toda pastilla o con el tableteo de una ametralladora. No le aseguramos nada, pero quizás la lectura de este artículo y una pizca de imaginación, así se lo permitan.



Durante los cuatro primeros capítulos de esta serie hemos recorrido bastante terreno, así que vamos a pararnos un momentito para recapitular lo que conocemos.

El comando SOUND es el que dice al **Ams-trad** que produzca un ruido. Y esto no es todo, además controla el tipo de sonido generado, lo que dura y su nivel.

La estructura más sencilla de dicho comando es:

SOUND canal, tono, duración, volumen

Puede oír el sonido que genere tecleando: SOUND,1,200,100,7

Comparándolo con la estructura básica, verá que el parámetro del canal es 1. El **Ams-trad** tiene tres canales de sonido: canal A, canal B y canal C. Cada uno puede sonar independientemente del otro, así que podemos conseguir que tres notas distintas suenen al mismo tiempo.

En el ejemplo anterior el parámetro de canal era 1, así que suena el canal A. Si hubiera sido 2, sonaría el canal B, mientras que el 4 seleccionaría el C.

Hoy bastantes más cosas sobre el parámetro del canal (volveremos sobre él la semana

que viene) pero por el momento nos quedaremos con los valores 1, 2 y 4 que seleccionan los canales A, B y C respectivamente.

El parámetro siguiente controla el tono, decidiendo si la nota ha de ser grave o aguda. Puede tomar valores comprendidos entre 0 y 4095.

Un parámetro con un valor grande hace que la nota producida sea más grave (*baja*). De la misma manera, un valor pequeño hace que sea más aguda (*alta*).

Cuando utilice el comando SOUND es necesario que le dé los parámetros de canal y tono. No podemos dejar de ponerlos como otros con los que nos encontraremos.

Duración y volumen: el sentido del ritmo

El parámetro de duración decide la longitud de la nota. Puede tomar valores enteros comprendidos entre 32767 y -32768.

Los números positivos fijan la duración de la nota medida en décimas de segundo. Por lo tanto un parámetro de 100 producirá una nota que suena durante un segundo, mientras que con 50 sonará medio segundo y con 1000 durante 10.

Los valores negativos del parámetro de duración tienen efectos diferentes. También determinan la longitud de la nota, pero esta vez con un método menos directo.

M. Barco





Cuando el número es negativo se le dice al **Amstrad** que la envolvente de volumen no sólo se utiliza para hacer variar la nota sino que además se repite un determinado número de veces. Si el parámetro duración es -3, la envolvente de volumen se repetirá tres veces. Si es -20 se repetirá veinte.

Al utilizar un parámetro de duración negativo, la longitud de la nota producida por el comando SOUND depende del número de veces que se repite la envolvente. Cada una dura un tiempo fijo, determinado por su propia definición.

De modo que si una envolvente de volumen en concreto dura dos segundos y el parámetro de duración es -3, la nota completa sonará durante seis segundos. La repetición de la envolvente determina la longitud total de la nota.

Volveremos sobre esto más adelante. El parámetro de volumen habla por sí mismo. Generalmente puede tener un rango de 0 (*silencio*) a 7 (*máximo nivel*) pero si se utiliza junto con una envolvente de volumen el rango puede llegar a alcanzar valores de 0 a 15.

No hay diferencia en el nivel máximo. Si no actúa una envolvente, 7 produce el mismo volumen que un parámetro 15 asociado a una envolvente. Es decir, con una envolvente de volumen podemos tener un control más fino sobre el nivel.

Podemos dejar sin especificar cualquiera de los parámetros de volumen o duración. Si no se le dan, el **Amstrad** asume que le estamos pidiendo una nota de volumen 4 con una duración igual a 20. Son los valores por defecto de los parámetros.

Ahora podemos comprender por que:
SOUND,4,200

es lo mismo que:

SOUND 4,200,20,4

Como ha podido ver, nuestro comando SOUND básico no es tan difícil de entender. En efecto, pudimos producir varios sonidos agradables usando sólo los cuatro parámetros que conocemos hasta ahora. De cualquier forma si lo que realmente desea es sacarle partido a todas las ventajas y habilidades del **Amstrad**, es conveniente que conozca algo más sobre las envolventes de tono y volumen.

Nuestro comando va a tomar la siguiente estructura:

SOUND conol, tono, duración, volumen, envolvente de volumen, envolvente de tono

Como ha podido ver hemos añadido dos parámetros más. El parámetro de la envolvente de volumen puede tomar valores entre 1 y 15. Estos números se refieren al de la definida previamente.

SONIDO

Un parámetro igual o 2 reproduciría el sonido de la envolvente de volumen número 2, mientras que un parámetro 5 llama a lo definido con el número 5. Estos envolventes de volumen determinan el nivel de la nota producida por un comando SOUND variándolo mientras suena.

Hay una envolvente de volumen 0. Es la que el comando toma por defecto y dura dos segundos como el volumen especificado en la instrucción SOUND.

Envolventes de volumen y sus parámetros

Hemos visto como SOUND llama a la envolvente de volumen. Vamos a echar un vistazo a cómo se definen. Se hace usando el comando ENV que, en su representación más sencilla toma la forma:

ENV,N,P,Q,R

PROGRAMAS

```
10 REM PROGRAMA 1
20 REM MAQUINA DE VAPOR
30 tono=1
40 FOR duracion=100 TO 20 STEP -10
50 SOUND 1,tono,duracion,7,0,0,1
60 SOUND 1,tono,duracion,7,0,0,15
70 NEXT duracion
80 WHILE NOT verdadera
90 SOUND 1,tono,duracion,7,0,0,1
100 SOUND 1,tono,duracion,7,0,0,15
110 WEND
```

N es el número que marca (*da número*) a la envolvente y puede tomar valores de 1 a 15. La misión de ésta es ir cambiando el nivel de una nota en un número fijo de escalones y cada uno de ellos dura un corto periodo de tiempo.

El parámetro P, que toma valores de 0 a 127, determina el número de escalones que tendrá la envolvente. El parámetro Q, cuyo valor estará entre -128 y 127, fija el cambio de volumen para cada uno de estos escalones. R elige la duración que tendrá cada paso, medido en décimas de segundo.

De esta forma definiendo una envolvente de volumen con:

ENV 12,5,2,100

Parámetro	Número N	Número de pasos en sección P	Cambio de volumen por paso Q	Longitud de cada paso R
Rango	1 o 15	0 o 127	-128 o 127	0 o 255

Tabla 1: Rangos de parámetro para el comando ENV.

se crea la número 12, que está formada por cinco escalones, cada uno de ellos con una duración de un segundo y que incrementan el volumen de dos unidades por cada escalón.

Seríamos capaces de escuchar su efecto con:

SOUND, 4,200,500,3,12

La tabla I nos muestra el rango de cada uno de los parámetros del comando ENV.

Probablemente recordará que en la definición que dimos anteriormente sobre la envolvente, sólo nos hemos referido a uno de sus secciones. De hecho podemos definir hasta cinco dentro de una envolvente de volumen y su fórmula correspondiente sería:

ENV N,P1,Q1,R1,P2,Q2,R2,
P3,Q3,R3,P4,Q4,R4,
P5,Q5,R5

Envolvente de tono y sus parámetros

La envolvente de tono es muy similar a la de volumen, excepto que afecta a lo grave o agudo que puede ser una nota, no a su nivel de volumen. Su estructura básica se define:

ENT S,T,V,W

Como podemos suponer S señala el número de envolventes, puede alcanzar valores entre 1 y 15. El valor que toma S por defecto es 0 y representa una envolvente que no cambia el tono.

El parámetro T determina el número de escalones que tienen la envolvente de tono. Sus valores están comprendidos entre 0 y 239.

El cambio de tono que hay entre cada uno de estos escalones viene dado por el parámetro V. Su rango de valores es de -128 a 127. Observe que cuando V es negativo el tono se hace más agudo en cada escalón mientras que con un valor positivo el tono se hace más grave.

El parámetro W da la longitud de cada escalón. Se mide, como es habitual, en décimos de segundo y es un número comprendido entre 0 y 255.

Así:

ENT 13,10,20,100

define la envolvente de tono número 13. Tiene 10 escalones y cada uno de ellos dura 1 segundo. En cada escalón se le suma 20 al parámetro de tono del comando SOUND. El resultado es una nota en tono descendente.

Escuche sus efectos con:

SOUND 2,500,1000,7,0,13

Lo mismo que la envolvente de volumen vista anteriormente, la de tono puede tener hasta cinco secciones, utilizando cada uno los parámetros ya citados.

Su definición más extensa es:

ENT S,T1,V1,W1,T2,V2,W2
T3,V3,W3,T4,V4,W4
T5,V5,W5

Parámetro	Número S	Número de pasos por sección T	Cambio de tono por paso V	Longitud de cada paso W
Rango	1 a 15 (-para repetición)	0 a 239	-128 a 127	0 a 255

Tabla II: Rangos de parámetro para el comando ENT.

La tabla II nos muestra el rango de los valores de los parámetros para el comando ENT.

Podemos hacer que la envolvente de tono se repita simplemente colocándola un número de envolvente negativo. Escuche el efecto que produce:

ENT -5,5,-20,100

sobre

SOUND, 1,600,2500,7,0,5

La envolvente de tono dura cinco segundos, repitiéndose mientras continúa el sonido. Observe que mientras el número dado en la definición es negativo, el comando SOUND continúa llamándola con un número positivo.

Dese cuenta también que, a diferencia de lo que ocurre con la envolvente de volumen, en este caso la repetición de lo de tono no influya sobre la duración del sonido.

Cuando llegamos al final del tiempo que dura el comando SOUND la nota termina, incluso si ha quedado a medio camino en una repetición. Escuche lo que ocurre cuando:

ENT -14,3,100,50

actúa de un modo riguroso sobre

SOUND 2,1000,400,7,0,14

Antes de nada, con el permiso de nuestras envolventes, le advertimos que por sí mismas ENT y ENV son mudas. No producen ni un solo ruido, sin embargo influyen sobre el sonido emitido por el comando SOUND.

Como puede ver por todo lo anterior, hemos recorrido un gran trayecto en estos cuatro artículos. Nuestra sencilla instrucción SOUND ha crecido y crecido. Ha pasado de tener cuatro parámetros —bastante obvios por cierto— a tener seis, con ENT ENV. ¡Hago un buen reparto con ellos!

De cualquier forma creemos que estará de acuerdo en que si los estudia paso a paso son bastante más sencillos de lo que parecen a simple vista.

El último parámetro

Y ahora añadamos un último parámetro a la instrucción SOUND. Es el que está colocado al final y nos permite hacer «ruido». Sí, no se extrañe. Es lo que hemos estado haciendo

durante estas últimas cuatro semanas, pero ahora se trata de un ruido diferente. Pruebe.

SOUND, 2,300,1000,7,0,0,13

y escuche sus efectos, no es lo que cabría esperar.

La explicación se encuentra en el último parámetro que hemos colocado al final del comando SOUND. Es el parámetro que hemos llamado muy acertadamente «ruido». Puede tomar valores comprendidos entre 1 y 15 (el 0 es el que tomo por defecto y no hace absolutamente nada) y añade una serie de sonidos al semi-azar a los notes que se podían esperar.

Nuestra instrucción SOUND tendrá ahora la siguiente apariencia:

SOUND canal, tono, duración,
volumen envolvente de volumen,
envolvente de tono, ruido

Intente hacer experimentos con este parámetro de «ruido» a ver que puede hacer. Utilice una nota como:

SOUND 1,200,200,7

y añádale ruido con líneas semejantes a:

SOUND 1,200,200,7,0,0,1

y

SOUND 1,200,200,7,0,0,15

Utilice todos los valores posibles para este parámetro «ruido» y con las envolventes de tono y de volumen así como con diferentes parámetros de tono, volumen, y duración. Se encontrará con toda clase de maravillosos efectos de sonido. (Envíenos por correo todos los que desee compartir. Gracias).

El programa I es nuestra particular versión de una máquina de vapor.

Y es todo por esta semana. Hemos resumido todos los parámetros de nuestro SOUND en la tabla III y le dejamos experimentando con sonido y ruidos.

Una última puntualización. ¿Recuerda que desde el primer artículo le hemos dicho que se deshaga de sus sonidos sobrantes introduciendo:

SOUND 129,100,0,0

en su Amstrad? ¿Qué pinta aquí un parámetro de canal igual a 129? Volveremos a ello la semana que viene.

	Canal	Tono	Duración	Volumen N		Envolvente de volumen	Envolvente de tono	Ruido
				Sin envolvente	Con envolvente			
Rango	1 - A	0	32767	0	0	1	1	1
	2 - B	a	a	0	0	0	0	0
	4 - C	4095	-32768	7	15	15	15	15
Defecto	Ninguna	Ninguna	20	4	12	0	0	0

Tabla III: Rango de parámetros para el comando SOUND.

GRATIS LAS TAPAS AL REALIZAR TU SUSCRIPCION

Ahora, al realizar tu suscripción, MICROHOBBY AMSTRAD te regala estas prácticas tapas especialmente diseñadas para contener tu revista favorita. • NO es necesario recurrir a ningún tipo de encuadernación ni manipulado.

• EN cualquier momento puedes separar un ejemplar determinado y volverlo a colocar en sólo unos segundos.

• SON prácticas y económicas...
... y con diseño especial para satisfacer a la gente inquieta.
(Oferta válida hasta el 31 de mayo de 1986)



Recorta o copia este cupón y envíalo a Hobby Press. Apartado de Correos 232. Alcobendas (Madrid).

Nombre _____ Apellidos _____
Dirección _____ C. Postal _____
Localidad _____ Provincia _____
Teléfono _____ Profesión _____

☐ Deseo suscribirme a **MICROHOBBY AMSTRAD** por un año (50 números) al precio de 6.400 ptas. (IVA incluido). Esta suscripción me da derecho a recibir, **totalmente gratis, las tapas** para contener la obra, valoradas en 670 ptas. (oferta válida hasta el 31 de mayo de 1986).

☐ Deseo recibir en mi domicilio las tapas de **MICROHOBBY AMSTRAD**, al precio de 670 ptas. (IVA incluido).

FORMA DE PAGO. MARCA CON UNA X LA OPCION QUE DESEES.

☐ Contra reembolso
☐ Mediante tarjeta de crédito VISA. Número de la tarjeta _____ Fecha caducidad de la tarjeta _____
☐ Mediante talón bancario o nombre de Hobby Press, S. A.
☐ Mediante giro postal n.º _____
☐ Mediante domiciliación bancaria
Banco _____ Sucursal y Localidad _____

Fecha y firma

N.º de cuenta _____

COMO ES, COMO SE HACE

¿Es usted político? ¿No...? ¿Y puede saberse a qué espera? Nosotros por nuestra parte le solucionamos un problema: Los discursos. Descubra la elocuencia y fluidez verbal de su Amstrad. Y lo que es más importante, todo se entiende pero nada se comprende.

por VOX POPULI



os vientos impetuosos que agitan las mesas y las masas electorales han callado poco a poco, las batallas dialécticas de nuestros insignes prohombres se han transmutado en una calma chi-

cha, presagio de nuevas y más feroces tormentas.

Los atónitos lectores que hayan sufrido en silencio la lectura del párrafo anterior, estarán de acuerdo con nosotros en que se nos ha colado en las páginas de la revista un trozo de discurso en el más puro estilo parlamentario.

Vamos a ser serios por una vez: esto es muy importante. Honradamente, **¿quién no se ha preguntado de dónde salen los discursos de**

los políticos y otras figuras públicas (con perdón)?

Estamos seguros que al público le encantaría averiguar dónde está la mágica fuente de la que se derraman esas parrafadas llenas de sentido y claridad, que permiten al hombre de la calle, estar al tanto sin problemas de la marcha de los asuntos de su país.

Pues bien, **Amstrad Semanal** lo ha conseguido. A través de líneas por las que sólo nosotros podemos transitar (*la 5, por ejemplo*), ponemos a nuestros lectores en bandeja el auténtico y genuino generador de discursos en su última versión, la 3.0, herramienta clave de uso común entre nuestros VIPS (*me refiero a las «very important persons»*).

Si, se trata de un programa, que, por supuesto, solamente corre en **Amstrad**.

Su lógica es muy sencilla: uno construye un grupo de frases sin sentido alguno, y las enlaza de tal forma que el resultado es una frase, más compleja, que tampoco tiene el menor significado, pero que, al oírse, se le caen a uno las lágrimas porque queda cantidad de bien.

El algoritmo empleado para hacer la selección de las frases que van a intervenir es el azar, en su forma más pura, y cualquiera que haya oído muchos discursos entenderá a qué me refiero.

¿Ejemplo? ¿Queréis ejemplos? Pues de eso nada. Recordad siempre que la ejemplificación indiscriminada de las estructuras formales de una semántica dada, causa una disgregación de lo global en lo parcial, y se sale completamente del contexto de cualquier casuística razonable.

No lo dude ni un momento, lance este programa al viento, y gánese el sustento en un Parlamento.



Amstravagancia

```

10 *****
20 AMSTRAVAGANCIAS
30 GENERADOR DE DISCURSOS
40 F.J.B.T. 1986
50 @ MICROHOBBY AMSTRAD
60 *****
70 *** INSTRUCCIONES
80 CLS:MODE 1:DIM a$(14),b$(14),c$(14),d$(14)
90 GOSUB 710
100 WINDOW #1,1,40,1,5
110 WINDOW #2,1,40,6,10
120 WINDOW #3,1,40,11,15
130 WINDOW #4,1,40,16,20
140 WINDOW #5,1,40,21,25
150 BORDER 25
160 PAPER #1,1:PEN #1,0:CLS #1
170 PAPER #2,2:PEN #2,0:CLS #2
180 PAPER #3,3:PEN #3,0:CLS #3
190 PAPER #4,8:PEN #4,1:CLS #4
200 PAPER #5,7:PEN #5,0:CLS #5
210 LOCATE #1,10,2:PRINT #1,"POR FIN EN SU AMSTRAD"
220 LOCATE #1,15,3:PRINT #1,"EL AUT ENTICO"
230 LOCATE #1,9,4:PRINT #1,"GENERADOR DE DISCURSOS"
240 LOCATE #2,2,1:PRINT #2,"Efectivamente. AMSTRAD y MICROHOBBY en estrecha colaboracion tecnica, han logrado alcanzar la cima del Software."
250 LOCATE #3,2,1:PRINT #3,"Y como primicia mundial, presentan el primer generador computerizado de discursos politicos del mundo."
260 LOCATE #4,2,1:PRINT #4,"En este ambicioso proyecto, AMSTRAD y MICROHOBBY han realizado la siguiente inversion en material.:"
270 LOCATE #5,2,1:PRINT #5,"8 horas de Programacion.
14 litros de Whisky escoces.
6 bolsas de cacahuetes.
5 kilos de patatas fritas."
280 FOR c=1 TO 10000:NEXT c
290 CLS #2:CLS #3:CLS #4:CLS #5:
300 LOCATE #2,10,1:PRINT #2,"INSTRUCCIONES DE USO
Pronunciase este discurso, en tono grandilocuente, con pausas de efecto."
310 LOCATE #3,2,1:PRINT #3,"No intentente mejorar la gramatica, pues esto le desmereceria mucho ante sus oyentes."
320 LOCATE #4,2,1:PRINT #4,"Al final, le resultara un esplendido discurso, valido para cualquier tema."
330 LOCATE #5,2,1:PRINT #5,"Quizá nadie le entienda, pero tampoco nadie podra discutir ni rebatir sus asertos."
340 FOR CO=1 TO 10000: NEXT CO
350 REM *** generador aleatorio
360 CLS #2:CLS #3:CLS #4: CLS #5:
370 x=1:GOTO 400
380 RANDOMIZE TIME:
390 x=ROUND(RND*12)+1
400 posa$=STR$(x)
410 IF LEN(refa$)>=27 THEN GOTO 660
420 IF INSTR(refa$,posa$)<>0 THEN 390
430 refa$=refa$+posa$
440 CLS #2:PRINT #2,a$(x);"...":
450 RANDOMIZE TIME:
460 y=ROUND(RND*12)+1
470 posb$=STR$(y)
480 IF INSTR(refb$,posb$)<>0 THEN 460
490 refb$=refb$+posb$
500 CLS #3:PRINT #3,b$(y);"...":
510 RANDOMIZE TIME:
520 z=ROUND(RND*12)+1
530 posc$=STR$(z)
540 IF INSTR(refc$,posc$)<>0 THEN 530

```

```

550 refc$=refc$+posc$
560 CLS #4:PRINT #4,c$(z);"...":
570 RANDOMIZE TIME:
580 w=ROUND(RND*12)+1
590 posd$=STR$(w)
600 IF INSTR(refd$,posd$)<>0 THEN 580
610 refd$=refd$+posd$
620 CLS #5:PRINT #5,d$(w);"...":
630 LOCATE #5,20,4: INPUT #5,"Pulse una tecla..."
640 GOTO 380
650 REM *** frases sueltas
660 CLS #2: PRINT #2,a$(14);"...":
670 CLS #3: PRINT #3,b$(14);"...":
680 CLS #4: PRINT #4,c$(14);"...":
690 CLS #5: PRINT #5,d$(14):
700 END:
710 a$(1) = " Queridos compañeros:"
720 a$(2) = " Por otra parte, y dado s los condicio- namientos actuales"
730 a$(3) = " Asimismo"
740 a$(4) = " Sin embargo, no hemos de olvidar que"
750 a$(5) = " De igual manera"
760 a$(6) = " La practica de la vida cotidiana nos prueba que"
770 a$(7) = " No resulta indispensable argumentar el peso y la significacion de estos problemas ya que"
780 a$(8) = " Las experiencias, ricas y diversas demuestran que"
790 a$(9) = " El afan de organizacion, pero, sobre todo"
800 a$(10) = " los superiores principios ideologicos, condicionan que"
810 a$(11) = " Incluso bien pudieramos atrevernos a sugerir que "
820 a$(12) = " Es obvio señalar que "
830 a$(13) = " Pero pecariamos de inciertos si soslayasemos que"
840 a$(14) = " Por ultimo, y como definitivo elemento esclarecedor, cabe añadir que"
850 b$(1) = "la realizacion de las pr emisas del programa"
860 b$(2) = "la complejidad de los estudios de los dirigentes"
870 b$(3) = "el aumento constante, en cantidad y en extension, de nuestra actividad"
880 b$(4) = "la estructura actual de la organizacion"
890 b$(5) = "el nuevo modelo de actividad de la organizacion"
900 b$(6) = "el desarrollo continuo de distintas formas de actividad"
910 b$(7) = "nuestra actividad de informacion y propaganda"
920 b$(8) = "el reforzamiento y desarrollo de las estructuras"
930 b$(9) = "la consulta con nuestros numerosos militantes"
940 b$(10) = "el inicio de la accion general de formacion de las actitudes"
950 b$(11) = "un relanzamiento especifico de todos los sectores implicados"
960 b$(12) = "la superacion de esperiencias periclitadas"
970 b$(13) = "una aplicacion indiscriminada de los factores confluyentes"
980 b$(14) = "el proceso consensuado de unas y otras implicaciones con corrientes"
990 c$(1) = "nos obliga a un exhaustivo analisis"
1000 c$(2) = "cumple un rol esencial en la formacion"
1010 c$(3) = "exige la precision y la determinacion"
1020 c$(4) = "ayuda a la preparacion y a la realizacion"

```

```

1030 c$(5) = "garantiza la participacion de un grupo importante en la formacion"
1040 c$(6) = "cumple deberes importantes en la determinacion"
1050 c$(7) = "facilita la creacion"
1060 c$(8) = "obstaculiza la apreciacion de la importancia"
1070 c$(9) = "ofrece un ensayo interesante de verificacion"
1080 c$(10) = "implica el proceso de reestructuracion y modernizacion"
1090 c$(11) = "habra de significar un autentico y eficaz punto de partida"
1100 c$(12) = "permite en todo caso explicitar las razones fundamentales"
1110 c$(13) = "asegura en todo caso un proceso muy sensible de inversion"
1120 c$(14) = "deriva en una directa incidencia superadora"
1130 d$(1) = "de las condiciones financieras y administrativas existentes"
1140 d$(2) = "de las directivas de desarrollo para el futuro"
1150 d$(3) = "del sistema de participacion general"
1160 d$(4) = "de las actitudes de los miembros hacia sus deberes ineludibles"
1170 d$(5) = "de las nuevas proposiciones"
1180 d$(6) = "de las direcciones educativas en el sentido del progreso"
1190 d$(7) = "del sistema de formacion de cuadros que corresponde a las necesidades"
1200 d$(8) = "de las condiciones de las actividades apropiadas"
1210 d$(9) = "del modelo de desarrollo"
1220 d$(10) = "de las formas de accion"
1230 d$(11) = "de las basicas premisas adoptadas"
1240 d$(12) = "de toda una casuística de amplio espectro"
1250 d$(13) = "de los elementos generados"
1260 d$(14) = "de toda una serie de criterios ideologicamente sistematizados en un frente comun de accion regeneradora."
1270 RETURN

```



Para que tus dedos no realicen el trabajo duro, M.H. AMSTRAD lo hace por ti. Todas las listas que incluyen este logotipo se encuentran a tu disposicion en un cassette mensual, solicítalo.

Sin duda alguna

A través de esta sección se pretende resolver, en la medida de lo posible, todas las posibles dudas que «**atormenten**» a todas las personas interesadas en el mundo del AMSTRAD, sean o no poseedores de uno y, si lo son, se encuentren en cualquier nivel de destreza en su manejo.

Semanalmente, aparecen en estas páginas las consultas de la mayor cantidad de usuarios posible; ello redundará en un mejor servicio y en un contacto más estrecho entre todos nosotros a través de la revista.

SIN DUDA ALGUNA está abierta a todos.

No todo es velocidad

Les agradecería que me contestaran a las siguientes preguntas:

1.ª Hace poco me he comprado un 6128 y no sé si tiene la función PAUSE como el SPECTRUM y otros. Si la respuesta es afirmativa me podrían decir cómo se puede hacer una pausa de 10 seg.

Oliver Ruiz. Murcia.

Simular una pausa en el **Amstrad** es muy sencilla, utilizando la función TIME, que nos devuelve el tiempo transcurrido desde que encendimos el ordenador en unidades de 1/50 de segundo. Bastará con teclear la siguiente línea hayá donde se deba de producir la pausa:

a = TIME: WHILE TIME < a + 50
* 10: WEND.

Esta línea provocará en el programa en el que esté incluido una pausa de 10 segundos.

Proceso por lotes CPM

Tengo un **Amstrad** 6128 y agradecería que me explicaran:

— para qué sirve y cómo se usa, la orden de CPM Plus SUBMIT;

— para qué se usa (también en CPM) el comando USER.

Javier Pereda Prado (Vizcaya)

La orden SUBMIT sirve, como explica y no detalla el manual, para el proceso por lotes en modo de consola. Lo más claro, creo, será realizar un pequeño ejemplo.

Entre en CPM, ya sabe, :CPM. Ahora lo primero que tenemos que hacer es editar un fichero tipo sub. Teclee ED prueba.sub y enter. ED es un editor de líneas que viene en el disco sistema. Para comenzar a editar el texto pulse, i+enter, ahora teclee lo siguiente:

DIR (pulse enter)

STAT (pulse enter)

Pulse ahora control+Z para finalizar la edición del fichero. Y finalmente, e y «enter» para grabar el texto en el disco. Si ahora introduce la orden SUBMIT PRUEBA, obtendrá el listado del directorio y la situación global del disco (STAT). Naturalmente, tanto SUBMIT.COM como STAT.COM, deberán encontrarse en el disco donde se encuentre PRUEBA.SUB. Lo que ha creado es un proceso por lotes, una especie de programita CPM, que le permitirá ejecutar todas las órdenes CPM que desee con una sola palabra.

Respecto a la otra pregunta, la orden USER sirve para cambiar de directorio. El disco se encuentra dividido en 10 partes numeradas desde la 0 a la 9 normalmente se trabaja en la 0, pero quizás buscando una mayor claridad prefiera utilizar estas particiones. Para utilizar la partición 3, por ejemplo, bastaría teclear:
USER 3

AMSDOS y CPM dos cosas distintas

Mis conocimientos de programación en Basic son muy escasos (fundamentalmente Sinclair) y nulos en **Amstrad**, por lo que no sé si mi duda es tal vez demasiado «tonta», pero ahí va:

En la propaganda del CPC 6128 leo que posee 128K de memoria RAM (41K usuario en Basic y 61K en CPM Plus).

Pero a la semana siguiente la publicidad no pone ya esta aclaración. ¿Esto qué significa?

Si son sumables ambas memorias salen 102K (de memoria RAM), ¿qué ocurren con el resto?

Y si no son sumables, ¿sólo se puede obtener hasta 61K de memoria máxima?

En definitiva quisiera saber cuántos K RAM de memoria útiles se pueden obtener con este aparato con las mejores técnicas de programación, pues cuando publicaron Vds. el artículo sobre el mismo en Banco de Pruebas, no encontré esta aclaración, tal vez por supuesta para un «entendido» en programación.

Fernando Guzmán Méndez (Palencia)

Como no hay una duda que tonta sea y sí mala información, ahí va nuestra explicación:

Esas dos memorias, 41 y 61, no son sumables puesto que se habla de cosas distintas. Una cosa es trabajar en AMSDOS, sistema operativo de nuestro CPC y que actúa desde que lo enciendes, y otra muy distinta es trabajar en CPM Plus o 2.2. Como por el momento trabajarás en Basic hay van los datos que te interesan.

El Basic por su constitución interna no puede gestionar más de 64K. Las «kas» que faltan desde 41 hasta 64 las utiliza el Basic para su gestión interna, firmware y área de pantalla. Evidentemente el segundo banco de memoria, 64K, te quedan libres para hacer lo que quieras, pero no en Basic. En el manual del CPC verás algunos ejemplos de su utilización.

Amstrad Ideas

AMSTRAD Semanal comunica a todos sus lectores la apertura de una nueva sección dedicada a recoger las mejores ideas que exploten al máximo las posibilidades del ordenador, materializadas en programas claros y cortos (*máximo 25 líneas*). Los mejores de entre todos ellos serán publicados con el nombre de su autor en la revista, recibiendo como premio, gratuitamente en su domicilio los cuatro primeros números de nuestra cinta mensual. Los programas enviados deberán incluir:

— **Cinta de cassette con el programa o programas grabados.**

— **Explicación detallada del funcionamiento y propósito del programa, mecanografiado a 2 espacios o con letra clara.**

Es imprescindible indicar en el sobre claramente: **AMSTRAD IDEAS.**

La dirección es:

Hobby Press, S. A.

La Granja, s/n.

Polígono Industrial de Alcobendas.
Madrid

ORDENMANIA SOFT

Características Contabilidad AMSTRAD PCW-8256 37.500 ptas.

- 1.—Posibilidad de abrir 500 cuentas y un total de 2.000 asientos, (por cada cara del disco).
- 2.—Posibilidad de reducir el número de cuentas y aumentar el de apuntes en la proporción 1/3 (cada cuenta no abierta admite 3 apuntes más). Ej: 250 cuentas admiten 2.750 apuntes, 50 cuentas, 3.500 apuntes, etc. (por cada cara de disco).
- 3.—Posibilidad de trabajar con cuentas de hasta 4 niveles de integración.
- 4.—Posibilidad de modificar o dar de baja apuntes integrados a mayor.
- 5.—Posibilidad de programar el balance de situación.
- 6.—Posibilidad de programar el cierre de la contabilidad.
- 7.—Posibilidad de programar las cuentas de explotación.
- 8.—Posibilidad de efectuar un cierre ficticio de la contabilidad.
- 9.—Posibilidad de renumerar los apuntes por fechas.
- 10.—Ejecución de balances comparativos.

TE OFRECEMOS EL NUEVO PLAN GENERAL CONTABLE CON I.V.A.

- 11.—Posibilidad de hacer de forma automática asientos dobles o múltiples.
- 12.—Acceso ultrarrápido para ejecución de extractos (tiempo de acceso medio a un asiento=0.5 segundos).
- 13.—Manejo fácil con menús interactivos para el usuario.
- 14.—Posibilidad de definir formato de página para impresora.
- 15.—Posibilidad de autogobernar un disco sin extraer para CONTINUAR la contabilidad en más de un disco.
- 16.—Posibilidad de relanzar balances comparativos por meses.
- 17.—Posibilidad de regeneración de la contabilidad.
- 18.—Posibilidad de programar conceptos automáticos.
- 19.—Dotado de medidas de seguridad para evitar pérdida de datos ante cortes de fluido eléctrico.
- 20.—Clave de acceso restringido a ciertas partes del programa (cierre de la contabilidad, borrado de discos, etc.).
- 21.—Posibilidad de hacer copias de seguridad de los ficheros al terminar la sesión.
- 22.—Servicio de Software postventa para atender dudas.
- 23.—Garantía ante fortuita degeneración del disco de programas.

Características Contabilidad AMSTRAD CPC-6128 y CPC-664 13.900 ptas.

- 1.—Creación de cuentas contables con límite máximo de 500 cuentas.

- 2.—Introducción de asientos, hasta un total de 1.000 como máximo.
- 3.—Modificación y cancelación de cuentas y asientos.
- 4.—Movimientos históricos de hasta 2.000 apuntes.
- 5.—Movimientos históricos de datos hasta 2.000 apuntes.
- 6.—Asientos simples o dobles, a su elección.
- 7.—Listador por pantalla o impresora.
- 8.—Libro diario, listados de cuentas, listado del PGC, balance de sumas y saldos, balance general de cuentas, balance de situación, cierre del ejercicio (Explotación, Resultados Extraordinarios, Pérdidas y Ganancias y Reparto de beneficios), Control del IVA así como todos los listados Históricos de Cuentas y Movimientos que desee efectuar.
- 9.—Ordenación de cuentas, Actualización de datos y descarga de movimientos, con toda la información Contable para la aplicación del IVA.

Disponemos de un equipo de software a tu servicio. Hacemos programas a medida.

RECUERDA: —Damos solución a la pequeña y mediana empresa.

Torres Quevedo, 34 Tel. 967/22 794
02003 ALBACETE

MICRO DEALER

AMSTRAD CENTER

Comandante Zorita, 13
Tel. 233 07 35 - 07 81
MADRID 28020

TODO EN:

- | | |
|-------------------|-------------------------|
| Joysticks | Impresoras |
| Cables | Interfaces |
| Monitores | Ampliaciones de memoria |
| Diskettes 3" | Teclados profesionales |
| Cassettes turbo | Cintas vírgenes |
| Unidades de disco | Diskette 5 1/4 DCDD |

MAYORISTAS DE:

- | | |
|-----------|--------------|
| Amstrad | Sinclair |
| Commodore | Spectravideo |
| Star | Newprint |
| Seikosa | Dk'tronics |

NOVEDAD

- Reserve su
Amstrad PCW8256
PVP 129.900

PRECIOS DE OFERTA EN IMPRESORAS:

20% Dto. sobre PVP
¡¡INFORMATE!!

¡¡BUSCAMOS DISTRIBUIDORES
EN PROVINCIAS!!

PIDA LISTA DE PRECIOS INFORMATE EN MICRO DEALER:

Tel. (91) 233 07 81
(915) 23 30 35



ESPECIALISTAS EN SOFTWARE COMERCIAL
Y A LA MEDIDA PARA
Y OTROS EQUIPOS **AMSTRAD**

** PROGRAMAS **

- ADMINISTRACION DE FINCAS
- GESTION INTEGRADA
(Facturación-Almacén-Clientes)
- ARCHIVO MULTIUSO
(Gimnasios-Academias-Tiendas de Discos-Bibliotecas, etc.)
- ETIQUETAS

- VIDEO-CLUBS
- CONTABILIDAD-FACTURACION
- PROGRAMAS TECNICOS
(Cálculos de estructuras: Ingeniero, Arquitectura, Andamios, Presupuestos y mediciones-Harmigán)
- CLINICAS (Podológicas-Veterinarias, etc.)

** SEMINARIOS ESPECIALIZADOS PARA TECNICOS **

** CLASES INDIVIDUALES (TUTORIAS) **

CONSULTE SIN COMPROMISO PRECIOS, EQUIPOS, TIEMPO DE REALIZACION, CURSOS, ETC.

ZURBANO, 4

☎ 410 47 63

28010 MADRID

Mercado común

Con el objeto de fomentar las relaciones entre los usuarios de AMSTRAD, **MERCADO COMUN** te ofrece sus páginas para publicar los pequeños anuncios que relacionados con el ordenador y su mundo se ajusten al formato indicado a continuación.

En **MERCADO COMUN** tienen cabida, anuncios de ventas, compras, clubs de usuarios de AMSTRAD, programadores, y en general cualquier clase de anuncio que pueda servir de utilidad a nuestros lectores.

Envíanos tu anuncio mecanografiado a: **HOBBY PRESS, S.A.**

AMSTRAD SEMANAL.

Aportado de correos 54.062
28080 MADRID

¡ABSTENERSE PIRATAS!

Vendo Amstrad CPC 464 con monitor en color, perfecto estado, regalo un lote de programas (Decathlon, Sabre-wulf... etc.), junto al manual y otros publicaciones en castellano y joystick. Llamar a Javier Medina al Tel. 27 12 15 de Las Palmas de G. C.

Vendo Amstrad CPC 664 fósforo verde. Comprado en agosto, prácticamente sin usar, 82.000 ptas. Tel. (91) 274 89 90, Alberto. Madrid.

Desearía contactar con usuarios del **Amstrad** (464, 664, 6128) para intercambio de juegos, información, ideas, listados, etc. José Oliveras. C/ Ganganell, 9. Besalu (Gerona). Tel. (972) 59 01 34.

Cambio Zx Spectrum 48K, 50 programas n.º 1 (Fairlight, Rambo, Yie Ar Kung Fu, Alien 8) por **Amstrad CPC 464**. Prestaría también bibliografía. Amador Merchán Ribera. C/ Cáceres, 8, 3.º A. 28045 Madrid. Tel. (91) 467 48 14.

Desearía cambiar **Amstrad CPC 464** con monitor en color por un **Amstrad CPC 6128** pagando diferencia. Tel. (91) 208 06 12, o mandando una carta a Javier Sánchez López. C/ Gálvez, 2, 2.º C. 28044 Madrid.

Vendo Amstrad CPC 464 con monitor en color y más de 30 programas entre juegos (Knight Lore, Alien 8, 3D-Voice Chess, Beach-Head, etc.), lenguajes (Pascal, Ensamblador, Desensamblador) y utilidades (procesadores de texto, hojas de cálculo, contabilidades, etc.). Regala joystick. Todo está en perfecto estado, 75.000 ptas. Tel. (91) 403 82 18.

Vendo Amstrad CPC 464, monitor fósforo verde con siete meses, precio 45.000 ptas, también unidad de disco con la garantía sin firmar por 45.000 ptas. (con disco CP/M y Logo y manual), conjunto por la cantidad de 85.000 ptas. Angel Díaz Sacristán. C/ Antonio López Torres, 53. Tel. (91) 764 61 08. 28033 Madrid.

Intercambio juegos arignales para **Amstrad CPC 464** en la ciudad de La Coruña. Llamar al Tel. (981) 23 53 84, o bien escribir a C/ Avda. Sardiñeira, 16, 4.º Izqda., y preguntar por Fco. Javier Neira Lamas.

Compro impresora compatible con **Amstrad 6128**. Preferible matricial o margarita con cinta de máquina. Tracción y fricción. Cádiz o Sevilla. Contacto: Tel. (956) 71 23 79. Alfonso.

Compraría joystick compatible con **Amstrad CPC 464**. Escribir detallando modelo y precio a Francisco M. Martín. C/ Cardenal Spínola, 31. Villaverde del Río (Sevilla).

Vendo monitor Zenith fósforo ocre, comprado en junio 85, nuevo por 19.200 ptas. Atari Video Computer Sistem con 10 juegos, información transformador, y 2 palancas compatibles Spectrum por 16.300 ptas. Alexis Gutiérrez. C/ Gutiérrez Rada, 2. Laredo (Cantabria). Envíos contra reembolso. Tel. (942) 60 62 25.

Cambio monitor en color con cables extendidos por monitor verde más cierta cantidad de dinero. Realizar las ofertas a: José Planelles Seguí. C/ Músico Peyró, 21, P-10. 46001 Valencia.

Vendo Amstrad CPC 464 con monitor de fósforo verde con aproximadamente 50 horas de utilización, por haber ganado mediante sorteo de una casa comercial un CPC 664 al mes y medio de adquirir el CPC 464. Incluiría además: 1 tratamiento de textos, 1 programa para ficheros, 1 programa de cálculos matemáticas, 1 programa para banco de datos, 1 curso didáctico para **Amstrad** en 2 cassettes, 1 juego Atom Smasher, 1 joystick PRO TM 3000, manuales en castellano. Todo ello por 63.000 ptas. Interesados dirigirse a: Antonio Ortiz García. C/ Avda. Portugal, 44 A. Malpartida de la Serena (Badajoz).

Vendo programas Decathlon y Superstart por ser incompatibles con mi ordenador. Interesados llamar al Tel. (958) 25 45 70 lunes, miércoles y viernes de 8 a 10 de la noche. Preguntar por Juan Santana.

MICRO-1

C/ Duque de Sesto, 50. 28009 Madrid
Tel.: (91) 275 96 16/274 53 80
(Metro O'Donell o Goya)

el IVA lo paga
MICRO-1

SOFTWARE: por cada programa GRATIS ¡¡1 BOLIGRAFO CON RELOJ DE CUARZO!!

HYPER SPORTS	2.300 ptas.
TORNADO LOW LEVEL	1.950 ptas.
EXPLODING FISTT	2.300 ptas.
JUMP JET	2.495 ptas.
ZORRO	2.600 ptas.
SABREWULF	1.650 ptas.
GHOSTBUSTERS	1.950 ptas.
GYROSCOPE	2.300 ptas.
HIGHWAY ENCOUNTER	1.750 ptas.
HIGHWAY ENCOUNTER DISCO	3.300 ptas.

DYNAMITE DAN	2.100 ptas.
RAID OVER MOSCOW	2.300 ptas.
THEY SOLD A MILLION	2.500 ptas.
FIGHTER PILOT	1.975 ptas.
MASTER OF T. LAMP	1.950 ptas.
NIGHTSHADE	1.950 ptas.
HACKER	1.950 ptas.
SUPER TEST	2.300 ptas.
MAPGAME	2.700 ptas.
TONADO LOW LEVEL DISCO	3.300 ptas.

JOYSTICK QUICK SHOTT II.. 2.295 ptas.
JOYSTICK QUICK SHOT V ... 2.595 ptas.

PC-COMPATIBLE IBM 256 K
MONITOR FOSFORO VERDE
2 BOCAS DISKETTE 360 K
SOLO ¡¡243.900!!

TAPA METACRILATO PARA
TECLADO ¡¡1.900 ptas.!!

UNIDAD DISKETTE 5.25"
¡¡45.900 ptas.!!
(incluido controlador)

LAPIZ OPTICO
¡¡4.900 ptas.!!

IMPRESORA MARGARITA
¡¡49.900 ptas.!!

CASSETTE ESPECIAL
ORDENADOR 5.295 ptas.

PRECIOS SUPER-EXCEPCIONALES PARA
AMSTRAD CPC-472 Y CPC-6128
¡¡LLAMANOS, TE ASOMBRARAS!!

IMPRESORAS
¡¡20% DTO. SOBRE P.V.P.!!

SINTETIZADOR DE VOZ
Y AMPLIFICADOR:
7.900 ptas.

MODULADOR TV
8.400 ptas.

CINTA C-15 ESPECIAL
ORDENADOR 85 ptas.
DISKETTE 3" 990 ptas.

UNIDAD DE DISCO 3" CON
CONTROLADOR: 49.900 ptas.

Libros:	
Curso autodidáctico Basic I	2.525 ptas.
Curso autodidáctico Basic II	2.525 ptas.
Programando con Amstrad	2.195 ptas.
Juegos sensacionales Amstrad	1.950 ptas.
Hacia la Inteligencia Artific.	1.295 ptas.
Música y sonidos con Amstrad	995 ptas.

NO PIERDA EL CONOCIMIENTO



etc

Avance en su propio mundo. Lea Alta Tecnología, la revista que, cada mes, le pone al día. Conozca los últimos avances que se producen en su campo de actividad: electrónica, genética, informática, comunicaciones, energía, transportes, nuevos materiales,

defensa... Las tecnologías de punta, aplicadas a su profesión y a la vida cotidiana, vistas con un enfoque interdisciplinario. Suscríbase ahora a Alta Tecnología. Aproveche la oferta de lanzamiento.

Alta Tecnología. La revista para personas con alto interés.

Edita TECNOLOGIA Y PRENSA, S. A. Arzobispo Morcillo, 24. Teléfono (91) 733 50 12. 28029 Madrid